

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-132587

(P 2 0 0 2 - 1 3 2 5 8 7 A)

(43) 公開日 平成14年5月10日 (2002.5.10)

(51) Int. Cl. ⁷	識別記号	F I	テーマコード (参考)
G06F 12/14	320	G06F 12/14	F 5B017
G11B 20/10		G11B 20/10	H 5C053
H04N 5/91		H04N 5/91	P 5D044

審査請求 未請求 請求項の数17 O L (全65頁)

(21) 出願番号	特願2000-320806 (P 2000-320806)	(71) 出願人	000002185 ソニー株式会社 東京都品川区北品川6丁目7番35号
(22) 出願日	平成12年10月20日 (2000.10.20)	(72) 発明者	秋下 徹 東京都品川区北品川6丁目7番35号 ソニ ー株式会社内
		(72) 発明者	石橋 義人 東京都品川区北品川6丁目7番35号 ソニ ー株式会社内
		(74) 代理人	100101801 弁理士 山田 英治 (外2名)

最終頁に続く

(54) 【発明の名称】 データ再生装置、データ記録装置、およびデータ再生方法、データ記録方法、並びにプログラム提供媒体

(57) 【要約】

【課題】 メディア、コンテンツの複数の異なる種類の識別子を持つリボケーションリストによる不正メディア、不正コンテンツの排除構成を実現するデータ処理装置を提供する。

【解決手段】 リボケーションリストに異なるカテゴリのメディア、コンテンツ双方の識別子を格納して、不正メディア、不正コンテンツ双方の排除を1つのリボケーションリストに基づいて実行可能な構成とした。リボーク対象ID、すなわちメディアID、コンテンツIDを1つのリボケーションリストに設け、デバイス側でメディア装着時にメディア識別子との照合、コンテンツ利用時にコンテンツ識別子との照合を行うことによって1つのリボケーションリストで複数の不正コンテンツ、不正メディアを排除する。

Revocation List ID	Revocation List Version	Number of Media1 ID	Media1 ID(0)	Media1 ID(N-1)	Number of Media2 ID	Media2 ID(0)	Media2 ID(N-1)	Number of Contents ID	Contents ID(0)	Contents ID(N-1)	ID of Revocation List
--------------------	-------------------------	---------------------	--------------	-------	----------------	---------------------	--------------	-------	----------------	-----------------------	----------------	-------	------------------	-----------------------

【特許請求の範囲】

【請求項 1】データ記憶手段に記憶されたコンテンツの再生処理を実行するデータ再生装置において、

データ記憶手段に対するアクセスを実行するメモリインタフェースと、該メモリインタフェースの制御を実行する制御部とを有し、

前記メモリインタフェースは、

処理禁止対象としたデータ記憶装置およびコンテンツのそれぞれの識別子データを持つリポケーションリストを格納した内部メモリを有し、

前記メモリインタフェースは、再生対象データを記録したデータ記憶手段からデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するとともに、

再生対象であるコンテンツの識別子を、データ記憶手段に記憶されたコンテンツのヘッダ情報から取得して、取得したコンテンツ識別子と、前記リポケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止する処理を実行する構成を有することを特徴とするデータ再生装置。

【請求項 2】前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする請求項 1 に記載のデータ再生装置。

【請求項 3】前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする請求項 1 に記載のデータ再生装置。

【請求項 4】前記メモリインタフェースは、前記制御部からのデータ再生要求コマンドに基づいて、再生対象コンテンツのヘッダ情報に格納されたコンテンツ識別子を取得し、該取得したコンテンツ識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする請求項 1 に記載のデータ再生装置。

【請求項 5】前記メモリインタフェースは、

外部から受領する更新用リポケーションリストについて、データ改竄チェック値（ICV）に基づくデータ改竄チェックを実行し、データ改竄なしの判定がなされ、かつ、

外部から受領する更新用リポケーションリストのバージョンと、前記内部メモリに格納済みのリポケーションリ

ストのバージョンとの比較処理を実行し、前記内部メモリに格納されたりポケーションリストのバージョンが、前記更新用リポケーションリストより新しいことが確認されたことを条件として前記更新用リポケーションリストによるリポケーションリストの更新処理を実行する構成を有することを特徴とする請求項 1 に記載のデータ再生装置。

【請求項 6】データ記憶手段に記憶するコンテンツの記録処理を実行するデータ記録装置において、

10 データ記憶手段に対するアクセスを実行するメモリインタフェースと、該メモリインタフェースの制御を実行する制御部とを有し、

前記メモリインタフェースは、

処理禁止対象としたデータ記憶装置およびコンテンツのそれぞれの識別子データを持つリポケーションリストを格納した内部メモリを有し、

20 前記メモリインタフェースは、データの記録対象となる前記データ記憶手段の識別子を受領し、該受領したデータ記憶手段の識別子と、前記リポケーションリストのリスト識別子との照合を実行し、識別子相互が一致する場合には、データ記録を中止する処理を実行する構成を有することを特徴とするデータ記録装置。

【請求項 7】前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする請求項 6 に記載のデータ記録装置。

30 【請求項 8】前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする請求項 6 に記載のデータ記録装置。

【請求項 9】データ記憶手段に記憶されたコンテンツの再生処理を実行するデータ再生方法であり、

前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、再生対象データを記録したデータ記憶手段からデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するとともに、

再生対象であるコンテンツの識別子を、データ記憶手段に記憶されたコンテンツのヘッダ情報から取得して、取得したコンテンツ識別子と、前記リポケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止する処理を実

行することを特徴とするデータ再生方法。

【請求項 10】前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リボケーションリストにリストされた識別子との照合処理を実行することを特徴とする請求項 9 に記載のデータ再生方法。

【請求項 11】前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リボケーションリストにリストされた識別子との照合処理を実行することを特徴とする請求項 9 に記載のデータ再生方法。

【請求項 12】前記メモリインタフェースは、制御部からのデータ再生要求コマンドに基づいて、再生対象コンテンツのヘッダ情報に格納されたコンテンツ識別子を取得し、該取得したコンテンツ識別子と、前記リボケーションリストにリストされた識別子との照合処理を実行することを特徴とする請求項 9 に記載のデータ再生方法。

【請求項 13】データ記憶手段に記憶するコンテンツの記録処理を実行するデータ記録方法であり、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、データの記録対象となる前記データ記憶手段の識別子を受領し、該受領したデータ記憶手段の識別子と、前記リボケーションリストのリスト識別子との照合を実行し、識別子相互が一致する場合には、データ記録を中止する処理を実行することを特徴とするデータ記録方法。

【請求項 14】前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リボケーションリストにリストされた識別子との照合処理を実行することを特徴とする請求項 13 に記載のデータ記録方法。

【請求項 15】前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リボケーションリストにリストされた識別子との照合処理を実行することを特徴とする請求項 13 に記載のデータ記録方法。

【請求項 16】データ記憶手段に記憶されたコンテンツの再生処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、再生対象データを記録したデータ記憶手段からデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リボケーションリストに

リストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するステップと、再生対象であるコンテンツの識別子を、データ記憶手段に記憶されたコンテンツのヘッダ情報から取得し、取得したコンテンツ識別子と、前記リボケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するステップと、を有することを特徴とするプログラム提供媒体。

10 【請求項 17】データ記憶手段に記憶するコンテンツの記録処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、データの記録対象となる前記データ記憶手段の識別子を受領し、該受領したデータ記憶手段の識別子と、前記リボケーションリストのリスト識別子との照合を実行し、識別子相互が一致する場合には、データ記録を中止するステップを、有することを特徴とするプログラム提供媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データ再生装置、データ記録装置、およびデータ再生方法、データ記録方法、並びにプログラム提供媒体に関する。特に、不正なコンテンツ処理の排除を目的として生成されるリボケーションリストによる不正メディアおよび不正コンテンツ双方の排除を可能としたデータ再生装置、データ記録装置、およびデータ再生方法、データ記録方法、並びにプログラム提供媒体に関する。

【0002】

【従来の技術】近年のインターネットの急激な普及、さらにモバイル型の小型再生器、ゲーム器等の普及に伴い、音楽データ、ゲームプログラム、画像データ等、様々なソフトデータ（以下、これらをコンテンツ (Content) と呼ぶ）の、インターネット等のネットワーク、あるいは、DVD、CD、メモ리카ード等の記憶媒体を介した流通が急増している。これらの流通コンテンツは、ユーザの所有する PC (Personal Computer)、再生専用器、あるいはゲーム機器においてネットワークから受信され記憶媒体に格納されたり、あるいはコンテンツを格納したメモ리카ード、CD、DVD等の記憶媒体を再生専用器、あるいはゲーム機器に装着することにより、コンテンツ再生処理、あるいはプログラム実行が可能となる。

【0003】コンテンツの記憶素子として、最近多く利用される素子にフラッシュメモリがある。フラッシュメモリは、EEPROM (Electrically Erasable Programmable ROM) と呼ばれる電氣的に書き換え可能な不揮発性メモリの一形態である。従来の EEPROM は、1 ビッ

トを2個のトランジスタで構成するために、1ビット当たりの占有面積が大きく、集積度を高くするのに限界があったが、フラッシュメモリは、全ビット一括消去方式により1ビットを1トランジスタで実現することが可能となった。フラッシュメモリは、磁気ディスク、光ディスク等の記録媒体に代わりうるものとして期待されている。

【0004】フラッシュメモリをデータ記録／再生機器に対して着脱自在に構成したメモリカードも知られている。このメモリカードを使用すれば、従来のCD（コンパクトディスク：登録商標）、MD（ミニディスク：登録商標）等のディスク状媒体に換えてメモリカードを使用するデジタルオーディオ記録／再生装置を実現することができる。

【0005】このような、フラッシュメモリを使用したコンテンツ記憶素子をパーソナルコンピュータ（PC）、再生器等において使用する場合、FAT（File Allocation Table）システムと呼ばれるファイル管理システムがアクセス情報テーブルとして一般的に使用される。FATシステムでは、必要なファイルが定義されると、その中に必要なパラメータがファイルの先頭から順番にセットされる。その結果、ファイルサイズを変長とすることができ、1ファイルを1または複数の管理単位（セクタ、クラスタ等）で構成することができる。この管理単位の関連事項がFATと呼ばれるテーブルに書かれる。このFATシステムは、記録媒体の物理的特性と無関係に、ファイル構造を容易に構築することができる。従って、FATシステムは、フロッピー（登録商標）ディスク、ハードディスクのみならず、光磁気ディスクにおいても採用することができる。上述したメモリカードにおいても、FATシステムが採用されている。

【0006】音楽データ、画像データ、あるいはプログラム等の様々なコンテンツは、再生機器として利用される再生装置、ゲーム機器、PC等の情報機器本体からのユーザ指示、あるいは接続された入力手段を介したユーザの指示により、上述のFATに基づいて例えば上述したフラッシュメモリから呼び出され、情報機器本体、あるいは接続されたディスプレイ、スピーカ等を通じて再生される。

【0007】さらに、ゲームプログラム、音楽データ、画像データ等、多くのソフトウェア・コンテンツは、一般的にその作成者、販売者に頒布権等が保有されている。従って、これらのコンテンツの配布に際しては、一定の利用制限、すなわち、正規のユーザに対してのみ、ソフトウェアの使用を許諾し、許可のない複製等が行われないようにする、すなわちセキュリティを考慮した構成をとるのが一般的となっている。

【0008】ユーザに対する利用制限を実現する1つの手法が、配布コンテンツの暗号化処理である。すなわち、例えばインターネット等を介して暗号化された音声

データ、画像データ、ゲームプログラム等の各種コンテンツを配布するとともに、正規ユーザであると確認された者に対してのみ、配布された暗号化コンテンツを復号する手段、すなわち復号鍵を付与する構成である。

【0009】暗号化データは、所定の手続きによる復号処理によって利用可能な復号データ（平文）に戻すことができる。このような情報の暗号化処理に暗号化鍵を用い、復号処理に復号鍵を用いるデータ暗号化、復号化方法は従来からよく知られている。

10 【0010】

【発明が解決しようとする課題】昨今、コンテンツ記録再生器において、不正なメディアや不正なコンテンツを排除するための方法としてリボケーションリストが提案されている。コンテンツの記録、再生を実行するデバイスは、例えばコンテンツ再生時にコンテンツを格納したコンテンツの識別子とリボケーションリストにリストされたコンテンツ識別子の照合を行ない、一致する識別子が見つかった場合は不正コンテンツであるとして再生処理を中止する処理を行なうことにより、不正なコンテンツの利用を排除することが可能となる。

20

【0011】しかし、従来、不正コンテンツ排除を目的とする場合は、不正なコンテンツ識別子を格納したリボケーションリストを用い、不正なメディアの排除を目的とする場合は、不正なメディア識別子を格納したリボケーションリストを用い、それぞれの用途に応じて参照するリボケーションを使い分けていた。この場合、デバイス側では、格納された複数のリボケーションリストから1つのリボケーションリストを選択する処理を必要とし、選択した後にコンテンツあるいはメディア識別子との照合を行なっていた。これらのリボケーションリスト選択処理は、新たなメディアの装着時、新たなコンテンツの処理時に繰り返し必要となり、処理の煩雑さを招く結果となる。

30

【0012】本発明は、このような処理の煩雑さを解消し、メディア、コンテンツの異なるカテゴリの識別子を1つのリボケーションに格納し、デバイスがリボケーションリストを選択することなく、メディアに対しても、コンテンツに対しても共通のリボケーションリストを適用して不正メディア、不正コンテンツの排除を行なうことを可能として処理の効率化を実現するデータ再生装置、データ記録装置、およびデータ再生方法、データ記録方法、並びにプログラム提供媒体を提供することを目的とする。

40

【0013】

【課題を解決するための手段】本発明の第1の側面は、データ記憶手段に記憶されたコンテンツの再生処理を実行するデータ再生装置において、データ記憶手段に対するアクセスを実行するメモリインタフェースと、該メモリインタフェースの制御を実行する制御部とを有し、前記メモリインタフェースは、処理禁止対象としたデータ

50

記憶装置およびコンテンツのそれぞれの識別子データを持つリポーションリストを格納した内部メモリを有し、前記メモリインタフェースは、再生対象データを記録したデータ記憶手段からデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するとともに、再生対象であるコンテンツの識別子を、データ記憶手段に記憶されたコンテンツのヘッダ情報から取得して、取得したコンテンツ識別子と、前記リポーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止する処理を実行する構成を有することを特徴とするデータ再生装置にある。

【0014】さらに、本発明のデータ再生装置の一実施態様において、前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする。

【0015】さらに、本発明のデータ再生装置の一実施態様において、前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする。

【0016】さらに、本発明のデータ再生装置の一実施態様において、前記メモリインタフェースは、前記制御部からのデータ再生要求コマンドに基づいて、再生対象コンテンツのヘッダ情報に格納されたコンテンツ識別子を取得し、該取得したコンテンツ識別子と、前記リポーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする。

【0017】さらに、本発明のデータ再生装置の一実施態様において、前記メモリインタフェースは、外部から受領する更新用リポーションリストについて、データ改竄チェック値（ICV）に基づくデータ改竄チェックを実行し、データ改竄なしの判定がなされ、かつ、外部から受領する更新用リポーションリストのバージョンと、前記内部メモリに格納済みのリポーションリストのバージョンとの比較処理を実行し、前記内部メモリに格納されたりポーションリストのバージョンが、前記更新用リポーションリストより新しいことが確認されたことを条件として前記更新用リポーションリストによるリポーションリストの更新処理を実行する構成を有することを特徴とする。

【0018】さらに、本発明の第2の側面は、データ記

憶手段に記憶するコンテンツの記録処理を実行するデータ記録装置において、データ記憶手段に対するアクセスを実行するメモリインタフェースと、該メモリインタフェースの制御を実行する制御部とを有し、前記メモリインタフェースは、処理禁止対象としたデータ記憶装置およびコンテンツのそれぞれの識別子データを持つリポーションリストを格納した内部メモリを有し、前記メモリインタフェースは、データの記録対象となる前記データ記憶手段の識別子を受領し、該受領したデータ記憶手段の識別子と、前記リポーションリストのリスト識別子との照合を実行し、識別子相互が一致する場合には、データ記録を中止する処理を実行する構成を有することを特徴とするデータ記録装置にある。

【0019】さらに、本発明のデータ記録装置の一実施態様において、前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする。

【0020】さらに、本発明のデータ記録装置の一実施態様において、前記メモリインタフェースは、前記制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合処理を実行する構成を有することを特徴とする。

【0021】さらに、本発明の第3の側面は、データ記憶手段に記憶されたコンテンツの再生処理を実行するデータ再生方法であり、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、再生対象データを記録したデータ記憶手段からデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するとともに、再生対象であるコンテンツの識別子を、データ記憶手段に記憶されたコンテンツのヘッダ情報から取得して、取得したコンテンツ識別子と、前記リポーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止する処理を実行することを特徴とするデータ再生方法にある。

【0022】さらに、本発明のデータ再生方法の一実施態様において、前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リポーションリストにリストされた識別子との照合処理を実行することを特徴とする。

【0023】さらに、本発明のデータ再生方法の一実施

態様において、前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行することを特徴とする。

【0024】さらに、本発明のデータ再生方法の一実施態様において、前記メモリインタフェースは、制御部からのデータ再生要求コマンドに基づいて、再生対象コンテンツのヘッダ情報に格納されたコンテンツ識別子を取

得し、該取得したコンテンツ識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行することを特徴とする。

【0025】さらに、本発明の第4の側面は、データ記憶手段に記憶するコンテンツの記録処理を実行するデータ記録方法であり、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、データの記録対象となる前記データ記憶手段の識別子を受領し、該受領したデータ記憶手段の識別子と、前記リポ

ーションリストのリスト識別子との照合を実行し、識別子相互が一致する場合には、データ記録を中止する処理を実行することを特徴とするデータ記録方法にある。

【0026】さらに、本発明のデータ記録方法の一実施態様において、前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段の識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行することを特徴とする。

【0027】さらに、本発明のデータ記録方法の一実施態様において、前記メモリインタフェースは、制御部からのメディア認識コマンドに基づいて、メディアであるデータ記憶手段との相互認証処理を実行し、該相互認証処理においてデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合処理を実行することを特徴とする。

【0028】さらに、本発明の第5の側面は、データ記憶手段に記憶されたコンテンツの再生処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、再生対象データを記録したデータ記憶手段からデータ記憶手段識別子を受領し、該受領したデータ記憶手段識別子と、前記リポケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するステップと、再生対象であるコンテンツの識別子を、データ記憶手段に記憶されたコンテンツのヘッダ情報から取得して、取得したコンテンツ識

別子と、前記リポケーションリストにリストされた識別子との照合を実行し、識別子相互が一致する場合には、データ再生処理を中止するステップと、を有することを特徴とするプログラム提供媒体にある。

【0029】さらに、本発明の第6の側面は、データ記憶手段に記憶するコンテンツの記録処理をコンピュータ・システム上で実行せしめるコンピュータ・プログラムを提供するプログラム提供媒体であって、前記コンピュータ・プログラムは、前記データ記憶手段に対するアクセスを実行するメモリインタフェースにおいて、データの記録対象となる前記データ記憶手段の識別子を受領し、該受領したデータ記憶手段の識別子と、前記リポケーションリストのリスト識別子との照合を実行し、識別子相互が一致する場合には、データ記録を中止するステップと、を有することを特徴とするプログラム提供媒体にある。

【0030】なお、本発明の第5、6の側面に係るプログラム提供媒体は、例えば、様々なプログラム・コードを実行可能な汎用コンピュータ・システムに対して、コンピュータ・プログラムをコンピュータ可読な形式で提供する媒体である。媒体は、CDやFD、MOなどの記録媒体、あるいは、ネットワークなどの伝送媒体など、その形態は特に限定されない。

【0031】このようなプログラム提供媒体は、コンピュータ・システム上で所定のコンピュータ・プログラムの機能を実現するための、コンピュータ・プログラムと提供媒体との構造上又は機能上の協働的關係を定義したものである。換言すれば、該提供媒体を介してコンピュータ・プログラムをコンピュータ・システムにインストールすることによって、コンピュータ・システム上では協働的作用が発揮され、本発明の他の側面と同様の作用効果を得ることができるのである。

【0032】本発明のさらに他の目的、特徴や利点は、後述する本発明の実施例や添付する図面に基づくより詳細な説明によって明らかになるであろう。

【0033】

【発明の実施の形態】〔システム概要〕図1に本発明のデータ処理装置の適用可能なコンテンツ配信システム構成を示す。例えば音楽データ、画像データ、その他各種プログラム等のコンテンツが、コンテンツ保持者またはサービスプロバイダのようなシステム運営者101から、インターネット等のネットワークを介して、またはCD、DVD、フラッシュメモリを搭載したメモ리카ード等の各種記録媒体であるメディア103に格納され、デバイス102に受信または装着されて再生、実行される。デバイスは、例えばパーソナルコンピュータ(PC)、再生専用器、ゲーム器等のコンテンツ再生機能を有するデバイスであり、例えば画像コンテンツを表示する表示装置、ユーザの指示を入力する入力装置を有する。

【0034】このようなコンテンツ配信システムの構成中、コンテンツを再生するデバイスと、コンテンツを格納するメディアとの詳細構成を図2に示す。

【0035】図2は、デバイス200、メディア1、210、メディア2、230の詳細構成を示している。メディア1、210は、単純なデータ読み出し、書き込み処理のみをサポートする制御部を持つメディアであり、メディア2、230は、メディアを装着するデバイスとの相互認証処理を実行し、またメディアに格納するコンテンツの暗号処理を実行するコントローラを有するメディアである。メディア1、210、メディア2、230の双方ともデバイス200に対する装着が可能である。

【0036】図2のデバイス200は、インターネット等のデータ通信手段を介したデータ送受信処理を実行する通信部201、各種指示を入力する入力部202、メッセージ、コンテンツ等の表示を実行する表示部203、これらの制御を実行する制御部205と、メディアとのデータ入出力処理のインタフェース機能を持つメモリインタフェース(I/F)部300とを持つデバイスコントローラ204、さらに、コンテンツのファイル群と、不正なメディアやコンテンツの失効情報としてのリボケーションリストを格納している内部メモリとしてのメモリ部207を有する。なお、内部メモリ内に格納されるリボケーションリスト等のデータファイルは、ファイル割り当てテーブルによって管理され読み出し可能な構成を持つ。

【0037】デバイス200は、コンテンツの再生時に再生対象のコンテンツがリボケーションリストに格納された失効メディア、失効コンテンツに対応していないことを確認した上で再生を行なう。再生対象のコンテンツがリボケーションリストにリストアップされていた場合は、再生エラーとなり、再生処理が実行されない。リボケーションリスト、およびリボケーションリストを適用した再生処理については後段で詳細に説明する。

【0038】メディア1、210は、データ入出力を制御する制御部211と、コンテンツを格納するメモリ部212を有し、メモリ部212は、コンテンツを対応ヘッダ情報とともに格納するのみならず、メディア個々に固有の識別情報としてのメディアID、さらに、メモリアクセスコントロール情報を記述したアクセス許可テーブルであるBPT(Block Permission Table)を格納している。

【0039】デバイス200のファイルシステムはメディアを認識した後に、アクセス許可テーブルであるBPTをメディアから読み込み、メディアへ直接アクセスを行うメモリインタフェース部300にBPTを転送し、管理させる。メモリインタフェース部300は、BPTを受信した後、受信したBPTについて改竄チェック値(ICV)の検証を行う。ICVが正当なものとして判断された場合のみ、BPTを有効なものとして保存す

る。メモリインタフェース部300は、メディアのメモリにアクセスする命令を受信した時、このメディアのBPTに基づいたアクセスのみ実行する。BPTの構成、およびBPTを用いた処理に関しては後段で詳細に説明する。

【0040】メディア2、230は、コントローラ231と、メモリ部232によって構成され、メモリ部232は、コンテンツを対応ヘッダ情報とともに格納し、さらにアクセス許可テーブルであるBPT(Block Permission Table)を格納している。コントローラ231は、メモリ部232に対するデータ格納、またはデータ読み出し用インタフェースとしてのメモリインタフェース(I/F)部234、メディアの識別子としてのメディアID、相互認証処理に適用する認証鍵Ka ke、コンテンツのメモリ部232への保存時の暗号鍵である保存鍵K s t o、さらに暗号化対象の鍵を暗号化する時の初期値I V _ k e y s等を格納した内部メモリ235、認証処理あるいはコンテンツの暗号化、復号処理を実行し、レジスタを備えた暗号処理部236、そして、これら各部の制御を実行する制御部233とを有する。

【0041】[メディア内メモリ構成]次に、メディア1、210、メディア2、230の各メモリ部のデータ格納構成を図3に示す。メモリ部は例えば、EEPROM(Electrically Erasable Programmable ROM)と呼ばれる電氣的に書き換え可能な不揮発性メモリの一形態であるフラッシュメモリであり、ブロック単位の一括消去方式によるデータ消去が実行される。

【0042】図3(a)に示すように、フラッシュメモリは、第1~Nまでの複数ブロックを有し、各ブロックは、(b)に示すように第1~Mまでの複数セクタによって構成され、各セクタは(c)に示すように実データを含むデータ部と、エラー訂正コード等の冗長データを含む冗長部によって構成される。後段で詳細に説明するが、冗長部には各セクタのデータ部内のセクタデータ改竄チェック値としてのICVが格納される場合がある。

【0043】[主要コマンド]次に図2のデバイス200において、制御部205と、メモリインタフェース(I/F)部300において発行される主なコマンドについて説明する。

【0044】まず、制御部205からメモリインタフェース(I/F)部300に対するコマンドには、以下のものがある。

・ステータス読み出しコマンド

現在のメモリインタフェース内のステータスを設定したステータスレジスタの状態の読み出し。メモリインタフェース(I/F)部300は、ステータスレジスタの内容を返す。

・セクタ読み出しコマンド

指定したセクタのデータ読み出し処理命令。

・セクタ書き込みコマンド

指定したセクタへのデータ書き込み処理命令。

・セクタ復号読み出しコマンド

セットされたヘッダの情報を元に、指定されたセクタの暗号化データを復号して読み出す処理の実行命令。

・セクタ暗号書き込みコマンド

セットされたヘッダの情報を元に、指定されたセクタへデータを暗号化して書き込む処理の実行命令。

・ヘッダ生成コマンド

指定されたパラメータを元にヘッダを生成する処理の実行命令。

・ヘッダセットコマンド

ヘッダをメモリーインターフェイス内にセットする処理の実行命令。

・BPTセットコマンド

BPTをメモリーインターフェイス内にセットする処理の実行命令。

・リボケーションリスト (Revocation List) セットコマンド

不正メディア、不正コンテンツのリストであるリボケーションリスト (Revocation List) をメモリーインターフェイス内にセットする処理の実行命令。

・更新用リボケーションリスト (Revocation List) チェックコマンド

更新用リボケーションリスト (Revocation List) に現在のリボケーションリスト (Revocation List) を更新してよいかチェックする処理の実行命令。

・メディア1認識コマンド

接続されたメディア1に対してメディアの識別子 (ID) を読み出して、そのIDが有効かどうかチェックする処理の実行命令。

・メディア2認識コマンド

接続されたメディア2に対して相互認証をして、メディアの識別子 (ID) が有効かどうかチェックする処理の実行命令。

・ファイル割り当てテーブル呼び出しコマンド

メモリ内のファイル割り当てテーブルを読み出す処理の実行命令。

・ファイル割り当てテーブル更新コマンド

メモリへのファイル割り当てテーブルを更新する処理の実行命令。

【0045】メモリーインターフェイス (I/F) 部300からメディア1に対するコマンドは、以下のものがある。

・ID読み出しコマンド

メディア1の持つIDを読み出す処理の実行命令。

【0046】〔デバイス内メモリーインタフェース詳細構成〕次にデバイス200のメモリーインタフェース (I/F) 部300の詳細構成を図4に示す。各構成部の機能を説明する。

【0047】・ステータスレジスタ301

メモリーインターフェイスの内部ステータスを保存するレジスタである。ステータスレジスタ301の構成例を図5に示す。各ビットは以下の意味を持つ。

・ビット0 (bit 0) : ビジーフラグ (1: ビジー (busy), 0: 待機 (ready))

メモリーインターフェイスが内部処理をしているかの判別用ビットである。

・ビット1 (bit 1) : 読み出し成功フラグ (1: 成功 (success), 0: 失敗 (fail))

10 メモリからデータの読み出しが成功したかの判別用ビットである。

・ビット2 (bit 2) : 書き込み成功フラグ (1: 成功 (success), 0: 失敗 (fail))

メモリヘッダの書き込みが成功したかの判別用ビットである。

・ビット3 (bit 3) : メディア1セットフラグ (1: セット (set), 0: 未セット (not set))

接続されたメディア1が利用可能かの判別用ビットである。

20 ・ビット4 (bit 4) : メディア2セットフラグ (1: セット (set), 0: 未セット (not set))

接続されたメディア2が利用可能かの判別用ビットである。

・ビット5 (bit 5) : メディア1有効フラグ (1: 有効 (OK), 0: 無効 (NG))

接続されたメディア1の識別子 (ID) が、リボケーションリスト (Revocation List) 内のリボーク (排除) メディア対象外かの判別用ビットである。

30 ・ビット6 (bit 6) : メディア2有効フラグ (1: 有効 (OK), 0: 無効 (NG))

接続されたメディア2の識別子 (ID) が、リボケーションリスト (Revocation List) 内のリボーク (排除) メディア対象外かの判別用ビットである。

・ビット7 (bit 7) : ヘッダセット成功フラグ (1: 成功 (success), 0: 失敗 (fail))

ヘッダがメモリーインターフェイス内にセット出来たかの判別用ビットである。

・ビット8 (bit 8) : ヘッダ生成成功フラグ (1: 成功 (success), 0: 失敗 (fail))

40 ヘッダの生成が成功したかの判別用ビットである。

・ビット9 (bit 9) : リボケーションリスト (Revocation List) セットフラ

グ (1: セット (set), 0: 未セット (not set))

リボケーションリスト (Revocation List) がメモリーインターフェイス内にセット出来たかの判別用ビットである。

・ビット10 (bit 10) : 更新用リボケーションリスト (Revocation List) 有効フラグ (1: 有効 (OK), 0: 無効 (NG))

50 更新用リボケーションリスト (Revocation List) が有

効であるかどうかの判別用ビットである。

【0048】ステータスレジスタ301は、これらのインタフェース(I/F)部300のステータス情報を保持する。

【0049】図4に戻り、各構成の機能について説明を続ける。

・コマンドレジスタ302

制御部より送信されたコマンドを保存するレジスタ

・アドレスレジスタ303

データの転送開始セクタを設定するレジスタ

・カウントレジスタ304

データの全転送セクタ数を設定するレジスタ

【0050】なお、外部メモリ、内部メモリに対するデータの読み書きは、アドレスレジスタに読み書きを開始するセクタアドレスを設定し、カウントレジスタに読み書きをする総セクタ数を設定し、コマンドレジスタにセクタ読み書きコマンドをセットすることで実行される。

【0051】・コントロールレジスタ305

メモリインターフェイスの動作を設定するレジスタ

・送受信制御部306

各種レジスタおよび送受信バッファなど、メモリインターフェイスの制御を行う。

・送信バッファメモリ307

送信データを格納するバッファ

・受信バッファメモリ308

受信データを格納するバッファ

・送信レジスタ309

送信バッファメモリ307内のデータを送信するためのレジスタ

・受信レジスタ310

受信したデータを保存し受信バッファメモリ308に転送するためのレジスタ

【0052】・暗号処理部320

送信バッファメモリ307、受信バッファメモリ308内のデータに対して、各種暗号処理を施す。

・メモリ部321

暗号処理部320における暗号処理に必要な鍵情報、および内部メモリから読み込まれるリボケーションリスト、外部メモリから読み込まれるアクセス許可テーブルとしてのブロック・パーミッション・テーブル(BPT)を格納、保存する領域である。リボケーションリスト、ブロック・パーミッション・テーブル(BPT)それぞれがメモリインタフェース内に有効にセットされた場合、送受信制御部306が制御部からのメディア認識コマンド、あるいは外部メモリに対するデータの読み書きコマンド等を受信した場合、セットされたりボケーションリスト、ブロック・パーミッション・テーブル(BPT)を参照した処理が実行される。これらの処理については、後段でフローを用いて詳細に説明する。

【0053】さらに、メモリ部321には、暗号処理に

必要な鍵情報としては、以下のデータが格納される。

Kdist:メディア2に格納されるコンテンツ以外のコンテンツのセキュリティヘッダ(Security Header)に含まれる配送鍵。コンテンツICV生成鍵Kicv_cont、コンテンツ鍵Kcを暗号化する。

Kicv_sh:セキュリティヘッダ(Security Header)のICVを生成する際に用いるセキュリティヘッダICV生成鍵。

IVsh:セキュリティヘッダ(Security Header)のICVを生成する際に用いる初期値(IV:Initial Value)。

MKake:相互認証用のマスター鍵。

IVake:相互認証用の鍵の生成処理に適用するための初期値(IV:Initial Value)。

IVauth:相互認証時のデータ生成用の初期値(IV:Initial Value)。

MKicv_rl:リボケーションリスト(Revocation List)のICV鍵を生成するマスター鍵。

IVicv_rl:リボケーションリスト(Revocation List)のICV鍵を生成する時の初期値(IV:Initial Value)。

IVrl:リボケーションリスト(Revocation List)のICV生成時に用いる初期値(IV:Initial Value)。

IV_keys:メディア2で、コンテンツ暗号化用の鍵を暗号化する時の初期値(IV:Initial Value)。

MKicv_bpt:アクセス許可情報であるBPT(Block Permission Table)のICV鍵を生成するマスター鍵。

IVicv_bpt:アクセス許可情報であるBPT(Block Permission Table)のICV鍵を生成する時のICV生成時に用いる初期値(IV:Initial Value)。

IVbpt:アクセス許可情報であるBPT(Block Permission Table)の初期値(IV:Initial Value)。

【0054】・ECC回路323

送信レジスタ309、受信レジスタ310にあるデータについて、ECCチェックを行う専用ブロックである。

【0055】・外部メモリ入出力インターフェイス324

外部メモリ(メディア1, 2)に対する入出力インターフェイス。外部メモリとしては例えばフラッシュメモリを搭載したメモリカード等がある。例えばコンテンツ、およびコンテンツ記録再生に伴うヘッダ情報、さらにブロック・パーミッション・テーブル(BPT)がこの外部メモリ入出力インターフェイスを介して入出力する。

・内部メモリ入出力インターフェイス325

内部メモリに対する入出力インターフェイス。当インタフェースを介して、内部メモリに格納された例えばリボケーションリストの入出力が実行される。

【0056】外部メモリ入出力インターフェイス32

4、および内部メモリ入出力インターフェイス325か

らは、処理に応じて以下の各信号が外部メモリ（メディア 1、2）、あるいは内部メモリに対して出力される。

CLE: コマンドラッチイネーブル

ALE: アドレスラッチイネーブル

CE: チップイネーブル

WE: ライトイネーブル

RE: リードイネーブル

また、外部メモリ（メディア 1、2）、あるいは内部メモリからの信号として、

WP: ライトプロテクト（外部メモリ（メディア 1、2）にのみ適用）

RDY/BUSY: レディー・ビジー

これら各種信号が入力される。

【0057】 [メモリ格納コンテンツ構成] 次に、メディアのフラッシュメモリに格納されるコンテンツ構成について図 6 を用いて説明する。音楽データ、画像データ等、各コンテンツは、図 6（a）に示すように各種属性情報からなるセキュリティヘッダと、実データ部としてのコンテンツとによって構成される。

【0058】 メディアのフラッシュメモリは、図 6（b）に示すように、複数コンテンツのセキュリティヘッダ部とコンテンツ部との各ペアを格納する。前述したように、フラッシュメモリはブロック単位で消去が実行されるので、1 ブロックには同一コンテンツに関するセキュリティヘッダ部またはコンテンツ部を格納する形態とし、一括した消去処理が許容される場合を除いて、異なるコンテンツを 1 つのブロックに格納する処理は行わない。

【0059】 [セキュリティヘッダ構成] セキュリティヘッダは、各コンテンツに対応する属性情報である。セキュリティヘッダのデータ構成を図 7 に示す。各データ内容について説明する。

【0060】 ・フォーマットバージョン (Format Version)

セキュリティヘッダ (Security Header) のフォーマットバージョンを示す。

・コンテンツ ID (Content ID)

コンテンツの識別子 (ID) を示す。

・コンテンツタイプ (Content Type)

コンテンツの種類を示す。例えばメディア 1、またはメディア 2 に格納されたコンテンツ、あるいは放送コンテンツ等である。

・データタイプ (Data Type)

コンテンツの属性、例えば音楽、画像等のデータであるか、プログラムであるか等を示す。

・暗号アルゴリズム (Encryption Algorithm)

コンテンツのコンテンツ鍵 (Kc) を使った暗号化処理アルゴリズムを示す。例えば DES による暗号化であるかトリプル DES (Triple-DES) によるか等を示す。

・暗号化モード (Encryption Mode)

暗号化アルゴリズム (Encryption Algorithm) で指定されたアルゴリズムに対応する暗号モードを示す。例えば ECB モードか CBC モードか等を示す。

【0061】 ・暗号化フォーマットタイプ (Encryption Format Type)

コンテンツの暗号化フォーマットを示す。タイプ 1 かタイプ 2 か、コンテンツ全体に対して 1 つのコンテンツ鍵 Kc で暗号化するタイプをタイプ 1 とし、コンテンツのセクタ毎に異なる鍵 Ksec_n を適用してコンテンツの暗号化を行なう態様をタイプ 2 とする。

【0062】 図 8 に各タイプの暗号化フォーマット構成を示す。図 8（a）がタイプ 1 の暗号化フォーマットで暗号化されたコンテンツのメモリ格納構成であり、

（b）がタイプ 2 の暗号化フォーマットで暗号化されたコンテンツのメモリ格納構成である。

【0063】 図 8（a）に示すタイプ 1 の暗号化フォーマットは、コンテンツがすべて 1 つのコンテンツ鍵 Kc を用いて暗号化されてメモリに格納された構成、すなわちセクタ非依存型暗号化処理である。図 8（b）に示すタイプ 2 の暗号化フォーマットは、フラッシュメモリの各セクタ毎に異なるセクタ鍵 Ksec_1 ~ Ksec_m が適用されて暗号化されたコンテンツが格納された構成、すなわちセクタ依存型暗号化処理である。例えば図 8（b）のフラッシュメモリのセクタ 1 では、セクタ 1 の暗号化鍵として Ksec_1 が対応して設定され、セクタ 1 に格納されるコンテンツは、各ブロックにおいて、すべて Ksec_1 を適用した暗号化処理が施されて格納される。フラッシュメモリのセクタ m では、セクタ m の暗号化鍵として Ksec_m が対応して設定され、セクタ m に格納されるコンテンツは、各ブロックにおいて、すべて Ksec_m を適用した暗号化処理が施されて格納される。

【0064】 このように、本発明の構成においては、各セクタ毎に異なる暗号化鍵を適用したコンテンツの暗号化処理が適用される。さらに、各セクタ毎に異なる暗号化鍵を適用した処理態様においても、1 つのセクタに 1 つの鍵を適用したシングル DES による処理、1 つのセクタに複数の鍵を適用したトリプル DES による処理等、各種の暗号化態様が適用可能である。これらの処理形態については、さらに後段で詳細に説明する。

【0065】 図 7 に戻り、セキュリティヘッダの構成について説明を続ける。

・暗号化フラグ (Encryption Flag)

ブロック内の各セクタの暗号化・非暗号化を示すフラグ。ブロック内のセクタ数（例えば 32 セクタ）分のフラグを持つ。例えば 0 : 非暗号化セクタ、1 : 暗号化セクタ。なお、本例では 1 ブロックを 32 セクタとする。

【0066】 ・ICV フラグ (ICV Flag)

ブロック内の各セクタの ICV 付加・非付加を示すフラグ。ブロック内のセクタ数（32 セクタ）分のフラグを持

つ。例えば 0 : I C V なし、1 : I C V あり

【0067】・暗号化コンテンツ鍵 (Kc_Encrypted 0-31)

暗号化されたコンテンツ鍵の格納領域(32個)

・暗号化 I C V 生成鍵 (Kicv_cont_encrypted)

暗号化されたコンテンツの I C V 作成のための鍵の格納領域

【0068】・有効リボケーションリストバージョン (Valid Revocation List version)

コンテンツ再生の際に有効に適用されるリボケーションリスト (Revocation List) のバージョン。コンテンツ再生の際に、セットされているリボケーションリスト (RevocationList) のバージョンがこれより古い場合、再生を許可しない。なお、自己デバイスにおいて格納したデータの再生処理等、リボケーションリストの参照を適用する必要がないコンテンツには 0 を設定する。

【0069】・セキュリティヘッダ I C V (ICV of Security Header)

セキュリティヘッダ (Security Header) の改竄チェック値 (I C V)。

【0070】 [リボケーションリスト] 次に、不正なメディアやコンテンツの失効情報としてのリボケーションリストの構成について説明する。図 9 にリボケーションリストの構成を示す。以下、各データについて説明する。

【0071】・リボケーションリスト識別子 (Revocation List ID)

リボケーションリスト (Revocation List) 固有の識別子としての I D である。

【0072】・リボケーションリストバージョン (Revocation List Version)

リボケーションリスト (Revocation List) のバージョンを示す。リボケーションリストは、更新され、更新時に新たな不正なメディアやコンテンツの失効情報を追加する。

【0073】本発明の構成では、リボケーションリスト (Revocation List) にバージョン情報を設定し、コンテンツのヘッダ内に有効なリボケーションリストのバージョン情報を設定する。コンテンツ読み出しの際に、現在デバイスに保持しているリボケーションリストのバージョンとコンテンツのヘッダ内にある有効なリボケーションリストのバージョンとを比較する。この際、現在保持しているリボケーションリストのバージョンの方がより古い場合には、コンテンツの読み出しを中止する。その結果、リボケーションリストの更新を行わなければ、そのコンテンツの読み出しはできない。

【0074】また、リボケーションリストの更新時にメモリ・インターフェース部が現在のリボケーションリストのバージョン情報と更新用のリボケーションリストのバージョン情報とを比較して、新しいリボケーションリ

ストであると判断した時のみ、リボケーションリスト更新を許可する構成とする。

【0075】バージョン情報を用いたリボケーションリストの新旧比較処理、更新処理の具体的処理例については、処理フローを用いて後段で詳細に説明する。

【0076】・メディア 1 I D 数 (Number of Media 1 ID)

失効しているメディア 1 (Media 1 ID) の総数

・メディア 1 I D (0) - メディア 1 I D (L-1)
(Media1ID(0) - Media1ID(L-1))

失効しているメディア 1 の識別子のリストである。

【0077】・メディア 2 I D 数 (Number of Media 2 ID)

失効しているメディア 2 (Media 2 ID) の総数

・メディア 2 I D (0) - メディア 2 I D (M-1)
(Media2ID(0) - Media2ID(M-1))

失効しているメディア 2 の識別子のリストである。

【0078】・コンテンツ I D 数 (Number of Contents ID)

20 失効しているコンテンツ I D (Contents ID) の総数

・コンテンツ I D (0) - コンテンツ I D (N-1)
(Contents ID(0) - Contents ID(N-1))

失効しているコンテンツ識別子のリストである。

【0079】・リボケーションリスト I C V (ICV of Revocation List)

リボケーションリストの改竄チェック用の I C V

【0080】上述のように、本発明におけるリボケーションリストは、複数の種類 (メディア、コンテンツ) の識別子 (I D) から構成される。このように、コンテンツやメディアの失効情報であるリボケーションリスト (Revocation List) に複数の種類のリボーク対象 I D、すなわちメディア I D、コンテンツ I D を設け、それぞれの照合を異なる動作として行うことによって、一つのリボケーションリストで複数のコンテンツ、メディアを排除することが可能となる。メディアの挿入時やコンテンツの読み出し時にメモリ・インターフェース部において、利用メディアまたは利用コンテンツの識別子 (I D) と、リボケーションリストにリストされた I D との照合を実行することにより、不正なメディアの使用や不正なコンテンツの読み出しを禁止することができる。

【0081】このようにコンテンツやメディアの複数の I D を一つのリボケーションリストに設定した構成により一つのリボケーションリストで複数の種類のメディアとコンテンツのリボーク (排除) が可能になる。メディア起動時のリボケーションリストに基づくメディアの検証処理、コンテンツ処理時のコンテンツ検証処理の具体的処理については、後段で説明する。

【0082】また、本発明の構成では、リボケーションリストは、外部メモリ等に直接アクセスするメモリイン

タフェースにセットアップされ、セットアップ後は、メディアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能な構成としたので、コンテンツの利用時に繰り返し内部メモリから読み出すなどの処理が不要となり処理が効率的に実行される。

【0083】 [ブロック・パーミッション・テーブル (BPT)] 次に、アクセス許可テーブルとして使用されるブロック・パーミッション・テーブル (BPT: Block Permission Table) の構成について説明する。従来、例えば PC 等においてコンテンツの再生を実行する場合、PC 内の OS のファイルシステムが主体的に、記録メディアに格納されているアクセス情報テーブル (例えば、File Allocation Table: FAT) を読み込んで管理しており、ファイルシステムがそのアクセス情報テーブルの内容を自由に書き換えが出来た。その為に、書き込み禁止を設定したアクセス情報テーブルを格納する記録メディアがあっても、そのアクセス情報テーブルをファイルシステムが読みとって書き換えることによって、記録メディア内のデータを書き換えられる可能性がある。

【0084】 本発明のデータ処理装置において採用されるブロック・パーミッション・テーブル (BPT) は、デバイスにおける書き替えを禁止したブロックに格納されるメディア自身のアクセス許可テーブルである。デバイスは BPT を格納したメディアを用いて、コンテンツデータ書き込み等のデータ処理を実行する場合、メディアに直接アクセスするデバイスのメモリインターフェイス部にブロック・パーミッション・テーブル (BPT) をセットすることで、デバイスの制御部がいかなるプログラムを実行中でも、メディアのアクセス許可テーブルであるブロック・パーミッション・テーブル (BPT) に設定された許可情報に従ったメモリアクセスが行われる構成とした。

【0085】 図 10 にブロック・パーミッション・テーブル (BPT) の構成を示す。以下、各データについて説明する。

【0086】 ・フォーマットバージョン (Format Version)

BPT (Block Permission Table) のフォーマットバージョンを示す。BPT 自体にも、各種のフォーマットがあり、そのいずれであるかを識別するデータである。

・BPT 識別子 (BPT ID)

ブロック・パーミッション・テーブル (BPT: Block Permission Table) の識別子 (ID) である。

・ブロック数 (Number of Blocks)

BPT (Block Permission Table) で扱うブロックの総数を示す。前述したように、フラッシュメモリはブロック毎の消去がなされる。BPT により管理されるブロック数を示している。

・ブロック # 1 - ブロック # n 許可フラグ (Block #1 -

#n Permission Flag)

各ブロックのアクセス制限フラグを示している。例えばフラグ 0 のブロックは、消去不可ブロックであり、フラグ 1 のブロックは消去可ブロックであることを示す。

・BPT-ICV (ICV of BPT)

BPT (Block Permission Table) の改竄チェック用の ICV である。

【0087】 デバイスのファイルシステムはデバイスを認識した後に、ブロック・パーミッション・テーブル (BPT) を例えばフラッシュメモリを搭載したメモリカード等のメディアから読み込み、メディアへ直接アクセスを行うメモリインターフェイス部に BPT を転送し、そのメディアに対するアクセス許可テーブルとして管理させる。メモリインターフェイス部は、アクセス許可テーブルを受信し BPT をセット (ex. 図 4 に示すメモリ部 321) する。メモリインターフェイスは、メディアのメモリにアクセスする命令を受信した時、このメディアのアクセス許可テーブルに基づいたアクセスのみを実行する。

【0088】 ブロック・パーミッション・テーブル (BPT) には、例えばメディアのフラッシュメモリの各ブロック単位での許可された処理態様、具体的には例えば消去可ブロック、消去不可ブロック、あるいは再生可ブロック、再生不可ブロック等の設定がなされている。メモリインタフェースは、これらの BPT 設定に従って処理の可否を決定する。これらの処理の詳細は、後段でさらに詳細に説明する。

【0089】 なお、ブロック・パーミッション・テーブル (BPT) には、改竄防止のための改竄チェック値 ICV が設定され、BPT のメモリインターフェイスへのセット時には、ICV チェックが実行され、改竄ありと判定された場合には、BPT のセット処理を実行しない。従って、不正なアクセス許可テーブルを作成して、使用することが防止される。BPT の ICV はメディアの識別子 (ID) に基づいて生成する。そのために、他のメディアにアクセス許可テーブルをコピーしたとしてもそのメディアは使用できない。ICV の生成については、後述する。

【0090】 メディアは、その製造時にブロック・パーミッション・テーブル (BPT) をメモリ (ex. フラッシュメモリ) の所定ブロックに書き込んで出荷する。この際、ブロック・パーミッション・テーブル (BPT) を格納したメモリ内のブロックについては、ブロック消去不可の設定をブロック・パーミッション・テーブル (BPT) に記述する。本発明のデバイスは、メディアに格納したデータ消去処理において、BPT を参照して BPT に設定された各ブロックの消去可否を参照した後、消去可であるブロックのみの消去を実行する構成であるので、BPT 格納ブロックを消去不可として設定したメディアについては、BPT の消去、書き換え替えが

防止される。メディア内のBPTを利用したファイルの書き込み、再生処理については後述する。

【0091】メディア（フラッシュメモリ搭載データ記録媒体）の製造時におけるブロック・パーミッション・テーブル（BPT）の設定フローを、図11および図12に示す。ここでは、メディアとコマンド通信が行えるメディア作成器を通してメディア識別子（ID）の生成とBPTの書き込みが連続動作で行われるものとする。

【0092】図11は、相互認証処理機能を持たないメディア1のタイプにおけるメディア作成器が実行するブロック・パーミッション・テーブル（BPT）の設定フローである。各処理について説明する。まず、まだ初期設定が行われていないメディアに対し、ID読み出しコマンドを送って（S31）、あらかじめメディアに格納されたIDを受信（S32）すると、そのIDをベースとしたICV生成鍵 K_{icv_bpt} を生成（S33）する。ICV生成鍵 K_{icv_bpt} は、マスター鍵： MK_{icv_bpt} と、初期値： IV_{icv_bpt} と、BPT識別子（ID）に基づいて生成する。具体的には、ICV生成鍵 $K_{icv_bpt} = DES(E, MK_{icv_bpt}, ID \cdot IV_{icv_bpt})$ に基づいて生成される。式の意味は、BPTのIDと初期値 IV_{icv_bpt} の排他論理和にマスター鍵： MK_{icv_bpt} によるDESモードでの暗号化処理を実行するという意味である。

【0093】次に、BPTの各フィールドに必要なパラメータを設定（S34）し、各パラメータが設定されたBPTに基づいてICVを生成（後述する図14の構成を適用）し（S35）、生成したICVをBPTのICVフィールドに設定（S36）する。このようにして構成されたブロック・パーミッション・テーブル（BPT）をメディア1に書き込む（S37）。なお、前述したようにBPTの書き込みブロックは、BPTにおいて消去不可領域として設定されたブロックとする。

【0094】図12は、相互認証処理機能を持つメディア2のタイプにおけるメディア作成器が実行するブロック・パーミッション・テーブル（BPT）の設定フローである。各処理について説明する。まず、まだ初期設定が行われていないメディア2との相互認証処理およびセッション鍵の共有（これらの処理については、後述する図22の処理を参照）を実行する。

【0095】相互認証および鍵共有処理が終了すると、メディア2に対しID読み出しコマンドを送って（S41）、IDを読み出し、IDをベースとしたICV生成鍵 K_{icv_bpt} を生成（S42）する。ICV生成鍵 K_{icv_bpt} は、マスター鍵： MK_{icv_bpt} と、初期値： IV_{icv_bpt} と、BPT識別子（ID）に基づいて生成する。具体的には、ICV生成鍵 $K_{icv_bpt} = DES(E, MK_{icv_bpt}, ID \cdot IV_{icv_bpt})$ に基づいて生成される。式の意味は、BPTのIDと初期値（ IV_{icv_bpt} ）の排他論理和にマスター

鍵： MK_{icv_bpt} によるDESモードでの暗号化処理を実行するという意味である。

【0096】次に、BPTの各フィールドに必要なパラメータを設定（S45）し、各パラメータが設定されたBPTに基づいてICVを生成（後述する図14の構成を適用）し（S46）、生成したICVをBPTのICVフィールドに設定（S47）する。このようにして構成されたブロック・パーミッション・テーブル（BPT）をメディア1に書き込む（S48）。なお、前述したようにBPTの書き込みブロックは、BPTにおいて消去不可領域として設定されたブロックとする。

【0097】図13にブロック・パーミッション・テーブル（BPT）の具体的構成例を示す。図13の（a）はメディア1、メディア2のフラッシュメモリのブロック構成であり、図13（b）は、ブロック・パーミッション・テーブル（BPT）である。ブロック・パーミッション・テーブル（BPT）は、フォーマット・バージョン、BPT ID、ブロック数に続いて、各ブロックの消去可（1）、消去不可（0）が設定され、最後にBPTの改竄チェック値（ICV of BPT）が格納された構成を持つ。メモリのBPT格納ブロック（図13の例ではブロック#2）は、ブロック・パーミッション・テーブル（BPT）において消去不可領域として設定され、デバイスによる消去を防止し、BPTの書き替えが実行されない構成を持つ。

【0098】なお、図13に示すブロック・パーミッション・テーブル（BPT）の構成例は、各ブロックの消去可（1）、消去不可（0）のみが設定された構成であるが、消去処理のみのアクセス許可を設定する構成ではなく、読み取り（再生）許可、不許可を設定した構成としてもよい。例えば再生および消去不可（11）、再生可、消去不可（10）、再生不可、消去可（01）、再生および消去可（00）とした設定が可能である。

【0099】なお、図2に示したようにメディア2ではメディア内に制御部231を持っており、ブロック・パーミッション・テーブル（BPT）が設定済みかどうかの状態を保持することもでき、BPTが設定されている状態で、デバイスからBPTの新たな書き込み命令が来たとしても、受け付けない構成として、BPTの再書き込みを防止する構成としてもよい。

【0100】なお、上述の例におけるBPT書き込みは、メディアとコマンド通信が行えるメディア作成器を通して実行する構成について説明したが、この他、メディアへのBPTの書き込みは、単純なメモリライターで作成したBPTを直接書き込む構成としてもよい。ただし、この場合も、メモリのBPT格納ブロックは、ブロック・パーミッション・テーブル（BPT）において消去不可領域として設定する。

【0101】〔改竄チェック値（ICV）による改竄チェック〕次に、改竄チェック値（ICV：Integrity Ch

eck Value) によるデータ改竄チェック処理について説明する。本発明の構成において、改竄チェック値(ICV)は、データ記憶手段に格納されるコンテンツ、ブロック・パーミッション・テーブル、リボケーションリスト等に付加され、それぞれのデータ改竄チェック処理に適用される。なお、コンテンツについての改竄チェック値は、セクタデータ単位に付加可能な構成である。コンテンツ、ブロック・パーミッション・テーブル、リボケーションリスト等に付加されたICV処理の具体的な形態については、後段で説明する。

【0102】DES暗号処理構成を用いた改竄チェック値(ICV)生成例を図14に示す。図14の構成に示すように対象となる改竄チェックデータを構成するメッセージを8バイト単位に分割(以下、分割されたメッセージをD0、D1、D2、・・・、Dn-1とする)する。改竄チェックデータは、例えばコンテンツ自体であったり、上述したアクセス許可テーブルであるBPTの構成データであったり、あるいはリボケーションリストの構成データである。

【0103】まず、初期値(Initial Value (以下、I Vとする))とD0を排他的論理和する(その結果をI 1とする)。次に、I 1をDES暗号化部に入れ、改竄チェック値(ICV)生成鍵K i c vを用いて暗号化する(出力をE 1とする)。続けて、E 1およびD1を排他的論理和し、その出力I 2をDES暗号化部へ入れ、改竄チェック値(ICV)生成鍵K i c vを用いて暗号化する(出力E 2)。以下、これを繰り返し、全てのメッセージに対して暗号化処理を施す。最後に出てきたE Nをコンテンツチェック値ICV'とする。

【0104】改竄のないことが保証された例えばコンテンツ生成時に生成した正当なICVと、新たにコンテンツに基づいて生成したICV'とを比較して同一性が立証、すなわちICV' = ICVであれば入力メッセージ、例えばコンテンツ、BPT、あるいはリボケーションリストに改竄のないことが保証され、ICV' ≠ ICVであれば改竄があったと判定される。

【0105】ICVを使用したデータ改竄チェック処理フローを図15に示す。まず、改竄チェックの対象データを抽出し(S11)、抽出したデータに基づいて例えば図14のDES暗号処理構成によりICV'を計算する(S12)。計算の結果、算出されたICV'とデータ内に格納されたICVとを比較し(S13)、一致した場合は、データの改竄が無く正当なデータであると判定(S14からS15)され、不一致の場合は、データの改竄があると判定(S14からS16)される。

【0106】リボケーションリストの改竄チェック用の改竄チェック値(ICV)生成鍵K i c v_r lは、予めデバイスのメモリインタフェース部300のメモリ部321(図4参照)内に格納されたリボケーションリスト(Revocation List)のICV鍵を生成するマスター

鍵:MKicv_rlと、リボケーションリスト(Revocation List)のICV鍵を生成する時の初期値:IVicv_rlと、リボケーションリストの属性情報中に含まれるリボケーションリスト・バージョン(Version)に基づいて生成する。具体的には、改竄チェック値(ICV)生成鍵K i c v_r l = DES(E, MKicv_rl, Version IVicv_rl)に基づいて生成される。前記式の意味は、バージョン(Version)と初期値(IVicv_rl)の排他論理和にマスター鍵:MKicv_rlによるDESモードでの暗号化処理を実行するという意味である。リボケーションリストの改竄チェック値は、このようにして生成されたICV生成鍵K i c v_r lを適用して初期値IVrl(メモリ部321に格納)を用いて図15に示すICV生成構成によって実行される。

【0107】また、ブロック・パーミッション・テーブル(BPT)の改竄チェック用の改竄チェック値(ICV)生成鍵K i c v_b p tは、予めデバイスのメモリインタフェース部300のメモリ部321(図4参照)内に格納されたBPTのICV鍵を生成するマスター鍵:MKicv_bptと、BPTのICV鍵を生成する時の初期値:IVicv_bptと、BPTの属性情報中に含まれるBPT識別子(ID)に基づいて生成する。具体的には、改竄チェック値(ICV)生成鍵K i c v_b p t = DES(E, MKicv_bpt, ID IVicv_bpt)に基づいて生成される。前記式の意味は、BPTのIDと初期値(IVicv_bpt)の排他論理和にマスター鍵:MKicv_bptによるDESモードでの暗号化処理を実行するという意味である。ブロック・パーミッション・テーブル(BPT)の改竄チェック値は、このようにして生成されたICV生成鍵K i c v_bptを適用して初期値IVbpt(メモリ部321に格納)を用いて図15に示すICV生成構成によって実行される。なお、BPTの付帯情報として格納されるICVは、BPT内のデータとBPTを格納したメディアの識別子(ID)を含むデータに基づいて生成される。従って、BPTのICVチェックは、BPTのデータ改竄の有無のみならず、メディア固有の正当なBPT、すなわち他のメディアにコピーされたBPTでないことを検証する機能も兼ね備える。

【0108】また、コンテンツのセクタ単位の改竄チェック用の改竄チェック値(ICV)生成鍵K i c v_c o n tは、コンテンツのヘッダ(セキュリティ・ヘッダ)中に暗号化されて格納されており、必要に応じてメモリインタフェースの暗号処理部320(図4参照)において、また、メディア2との相互認証後に実行されるメディア2のコントローラ231で実行されるDES-CBCモードによる復号処理によって取得される。これらの処理についてはフローを用いた説明中で詳細に説明する。

【0109】このようなデータ改竄チェックの結果、例えばリボケーションリストの改竄が明らかになれば、リ

ボケーションリストの参照処理に基づくコンテンツの再生等の処理を禁止し、また、アクセス許可テーブルであるBPTに改竄があると判定されれば、BPTに基づくメディアのデータに対するアクセスを禁止する処理を実行する。これらの処理については、後段で詳細に説明する。

【0110】【データ読み出し、書き込み処理】以下、本発明のデータ処理装置において、デバイスがメディアからのデータ読み出しを行なう場合の処理、およびデバイスがメディアに対してデータを格納する場合に実行される処理について説明する。

【0111】（デバイス起動時処理）まず、デバイスを起動させた場合における処理を図16を用いて説明する。図16は、左側に図2におけるデバイス200の制御部205の処理、右側にメモリインタフェース部300の処理を示したものである。処理スタート時点でのメモリインタフェース部300のステータスレジスタの状態は、ビジーフラグ：0（待機）、リボケーションリストセットフラグ：0（未セット）である。

【0112】まず、デバイスが起動すると、制御部は、内部メモリのファイル割り当てテーブル呼び出しコマンドをメモリインタフェース部に送信（S101）する。メモリインタフェース部は、デバイスの内部メモリに対してファイル割り当てテーブルの読み出しコマンドを送信（S102）して、ファイル割り当てテーブルを内部メモリから受信し、制御部に送信（S103）する。

【0113】なお、ファイル割り当てテーブルは、デバイスのアクセス可能な内部メモリ、外部メモリに格納されたデータ、例えば様々なコンテンツ、あるいはリボケーションリスト等、各種データファイルをディレクトリ管理するテーブルであり、例えば図17に示すように、ディレクトリ、ファイル名、格納セクタが対応付けられた構成を持つ。デバイスは、ファイル割り当てテーブルに基づいて、様々なファイルのアクセスを行なう。

【0114】制御部は、内部メモリに格納されたデータに対応するファイル割り当てテーブルを受信（S104）すると、テーブルに基づいてリボケーションリストの読み出し処理を実行（S105）し、リボケーションリストのセットコマンドと、リボケーションリストをメモリインタフェース部に送信（S106）する。リボケーションリストのセット処理は、リボケーションリストが有効である場合にのみ実行され、リストがセットされると、メディアからのコンテンツ読み出し処理等、コンテンツ処理の際、リボケーションリストにリストアップされたコンテンツまたはメディア識別子との比較処理を実行する。これらの処理については後述する。

【0115】リボケーションリストのセットコマンドと、リボケーションリストを制御部から受信（S107）すると、メモリインタフェースは、ステータスレジスタのビジーフラグを1（ビジー）にセット（S10

8）し、リボケーションリストの改竄チェック用の改竄チェック値（ICV）生成鍵Kicv_rlを生成（S109）する。

【0116】リボケーションリストの改竄チェック用の改竄チェック値（ICV）生成鍵Kicv_rlは、予めデバイス内に格納されたリボケーションリスト（Revocation List）のICV鍵を生成するマスター鍵：MKicv_rlと、リボケーションリスト（Revocation List）のICV鍵を生成する時の初期値：IVicv_rlと、リボケーションリストの属性情報に含まれるリボケーションリスト・バージョン（Version）に基づいて生成する。具体的には、改竄チェック値（ICV）生成鍵Kicv_rl=DES（E, MKicv_rl, Version, IVicv_rl）に基づいて生成される。式の意味は、バージョン（Version）と初期値（IVicv_rl）の排他論理和にマスター鍵：MKicv_rlによるDESモードでの暗号化処理を実行するという意味である。

【0117】次にメモリインタフェースは生成した改竄チェック値（ICV）生成鍵Kicv_rlを用いてリボケーションリストのICV'を生成し、予めリボケーションリスト内に格納された正しいICVとの照合処理（ICV' = ICV?）を実行（S110）する。なお、ICV'の生成処理は、前述の図14で説明したDESモードに基づいて、初期値IVrlを用い、生成した改竄チェック値（ICV）生成鍵Kicv_rlを適用した処理によって行われる。

【0118】ICV' = ICVである場合（S111でYes）は、リボケーションリストが改竄のない正当なものであると判定され、コンテンツの読み出し処理等の際に参照可能な状態にセットし、リボケーションリストセットフラグを1（セット）にセット（S112）する。リボケーションリストはメモリインタフェース内のメモリ（例えばメモリ部321（図4参照））に格納され、例えば、送受信制御部306が制御部205（図2参照）からメディア認識コマンドを受信するとセットされたリボケーションリストのメディア識別子と、デバイスに装着したメディアのメディア識別子との照合が実行され、また、送受信制御部306が制御部205からコンテンツの読み出し処理に伴うヘッダセットコマンドを受信するとセットされたリボケーションリストのコンテンツ識別子と、読み出し対象コンテンツのコンテンツ識別子との照合が実行される。

【0119】このように、リボケーションリストは、外部メモリ等に直接アクセスするメモリインタフェースにセットアップされ、セットアップ後は、メディアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能な構成とされ、コンテンツの利用時に繰り返し内部メモリから読み出すなどの処理が不要となり処理が効率的に実行される。

【0120】図16のフローの説明を続ける。ICV'

≠ I C Vである場合 (S 1 1 1でN o) は、リボケーションリストに改竄ありと判定され、リストの参照処理に基づくコンテンツ処理を禁止し処理を終了する。以上の処理の終了により、ビジーフラグは0にセットされる。

【0121】一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信 (S 1 1 4) し、ビジーフラグが0となったことを条件 (S 1 1 5) とし、リボケーションリストセットフラグを保存 (S 1 1 6) する。保存されるリボケーションセットフラグは、リストの改竄が無いと判定された場合は、リストが有効にセットされたことを示す1、その他の場合は0となる。

【0122】 (メディア認識時処理) 次に、デバイスにメディアが装着された場合のメディアの有効性確認等、メディア認識時に実行する処理について説明する。前述したようにメディアには、デバイスとの相互認証処理を実行しないタイプのメディア1と、デバイスとの相互認証処理を実行するタイプのメディア2とがある。デバイスは、それぞれのタイプがデバイスに装着されると、メディアを利用したコンテンツ処理を実行してよいか否か、具体的にはリボケーションリストに不正メディアとしての登録がないかを確認する処理を実行し、装着メディアがリボケーションリストにリストアップされておらず、有効に利用可能なメディアであることが確認されたことを条件として、メディアに格納されたアクセス許可テーブルであるBPT (Block Permission Table) をメモリインタフェースにセットし、BPTを参照したメモリアccessを可能とする処理を実行する。

【0123】まず、メディア1が装着された場合のメディア確認処理について図18、図19を用いて説明する。

【0124】図18、図19においても左側に図2におけるデバイス200の制御部205の処理、右側にメモリインタフェース部300の処理を示している。当フロー開始時点で、メモリインタフェース部300のステータスレジスタの状態は、ビジーフラグ: 0 (待機)、メディア1有効フラグ: 0 (無効)、メディア1セットフラグ: 0 (未セット) の状態である。

【0125】まず、制御部は、デバイスに装着されたメディアがメディア1であることを認識する (S 2 0 1) 。メディア識別は予め設定されたメディア形状に基づく機械的情報あるいはデバイス、メディア間の通信情報に基づいて行われる。制御部がメディア1であることを認識すると制御部は、メディア1認識コマンドをメモリインタフェースに送信する (S 2 0 2) 。

【0126】メモリインタフェースは、制御部からのメディア1認識コマンドを受信 (S 2 0 3) すると、ステータスレジスタのビジーフラグを1 (ビジー) に設定し (S 2 0 4) 、メディア1に対してメディア1の識別子 (I D) の読み出しコマンドを送信 (S 2 0 5) し、受

信 (S 2 0 6) する。さらに、受信したメディア1のI Dと、既にセットされているリボケーションリスト中のリボーク (排除) メディア1のリストとの比較照合を実行 (S 2 0 7) する。リボケーションリストは、先の図16の起動時フローにおいて説明したように、起動時にメモリインタフェースにセットアップされ、セットアップ後は、メディアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能となる。

【0127】受信I Dと一致するI Dがリスト中に存在しなかった場合は、装着メディア1はリボーク対象メディアではなく、有効に利用可能なメディアであると判定 (S 2 0 8においてN o) し、ステータスレジスタのメディア1有効フラグを1 (有効) にセット (S 2 0 9) し、ビジーフラグを0 (待機) にセット (S 2 1 0) する。受信I Dと一致するI Dがリボケーションリスト中にあった場合 (S 2 0 8においてY e s) は、装着メディア1はリボーク対象メディアであり、有効に利用できないと判定し、ステップS 2 0 9の有効フラグの有効化処理を実行せずステップS 2 1 0でビジーフラグを0 (待機) にセットして処理を終了する。

【0128】一方、制御部は、ステップS 2 1 1において、ステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが0 (待機) になったことを確認 (S 2 1 2) の後、メディアフラグ状態を確認して有効 (フラグ: 1) である場合 (S 2 1 3でY e s) にのみ処理を続行し、無効 (フラグ: 0) である場合 (S 2 1 3でN o) は、処理を終了する。

【0129】次に、図19に進み、制御部は、メディア1に関するファイル割り当てテーブル呼び出しコマンドをメモリインタフェースに送信 (S 2 2 1) し、メモリインタフェースは、ファイル割り当てテーブルの格納されたセクタ読み出しコマンドをメディア1に送信 (S 2 2 2) し、ファイル割り当てテーブルをメディア1から受信し、制御部に送信 (S 2 2 3) する。

【0130】制御部は、メディア1に格納されたデータに対応するファイル割り当てテーブルを受信 (S 2 2 4) すると、テーブルに基づいてブロック・パーミッション・テーブル (BPT) の読み出し処理を実行 (S 2 2 5) し、BPTのセットコマンドと、BPTをメモリインタフェースに送信 (S 2 2 6) する。BPTのセット処理は、BPTが有効である場合にのみ実行され、BPTがセットされると、メディアからのコンテンツ書き込み処理等、コンテンツ処理の際、BPTを参照してブロック毎の消去が可能か否かを判定する。実際のBPTを参照したデータ書き込み処理については、後段で説明する。

【0131】ブロック・パーミッション・テーブル (BPT) のセットコマンドと、BPTを制御部から受信 (S 2 2 7) すると、メモリインタフェースは、ステー

タスレジスタのビジーフラグを1 (ビジー) にセット (S228) し、BPTの改竄チェック用の改竄チェック値 (ICV) 生成鍵 K_{icv_bpt} を生成 (S229) する。

【0132】BPTの改竄チェック用の改竄チェック値 (ICV) 生成鍵 K_{icv_bpt} は、予めデバイス内に格納されたBPTのICV鍵を生成するマスター鍵: MK_{icv_bpt} と、BPTのICV鍵を生成する時の初期値: IV_{icv_bpt} と、メディアIDに基づいて生成する。具体的には、改竄チェック値 (ICV) 生成鍵 $K_{icv_bpt} = DES(E, MK_{icv_bpt}, \text{メディアID} \parallel IV_{icv_bpt})$ に基づいて生成される。式の意味は、メディアIDと初期値 (IV_{icv_bpt}) の排他論理和にマスター鍵: MK_{icv_bpt} によるDESモードでの暗号化処理を実行するという意味である。

【0133】次にメモリインタフェースは生成した改竄チェック値 (ICV) 生成鍵 K_{icv_bpt} を用いてBPTのICV' を生成し、予めBPT内に格納された正しいICV値との照合処理 ($ICV' = ICV?$) を実行 (S230) する。なお、ICV' の生成処理は、前述の図14で説明したDESモードに基づいて、初期値 IV_{bpt} を用い、生成した改竄チェック値 (ICV) 生成鍵 K_{icv_bpt} を適用した処理によって行われる。なお、BPTの付帯情報として格納されたICVは、メディアIDを含むデータに基づいて生成されており、ICVのチェックは、BPTのデータ改竄の有無のみならず、メディア固有の正当なBPT、すなわち他のメディアにコピーされたBPTでないことの検証も兼ね備える機能を持つ。

【0134】 $ICV' = ICV$ である場合 (S231でYes) は、BPTが正当なメディアに格納された改竄のない正当なものであると判定され、コンテンツ処理等の際に参照可能な状態にセットし、メディア1セットフラグを1 (セット) にセット (S232) する。 $ICV' \neq ICV$ である場合 (S231でNo) は、BPTに改竄ありと判定され、BPTの参照処理に基づくコンテンツ処理を禁止し処理を終了する。以上の処理の終了により、ビジーフラグは0にセット (S233) される。

【0135】一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信 (S234) し、ビジーフラグが0となったことを条件 (S235でYes) としてメディア1セットフラグを保存 (S236) する。保存されるメディア1セットフラグは、BPTの改竄が無いと判定された場合は、メディア1が有効にセットされたことを示す1、その他の場合は0となる。

【0136】次にメディア2がデバイスに装着された際のメディア2確認処理について、図20、図21を用いて説明する。メディア2は、図2を用いて説明したように、デバイスとの相互認証を実行するメディアである。

【0137】図20のステップS301からS304のステップは、メディア1の確認処理におけるステップS201~S204と同様であるので説明を省略する。

【0138】ステップS305において、メモリインタフェースは、メディア2との相互認証処理を実行する。

【0139】図22に、共通鍵暗号方式を用いた相互認証方法 (ISO/IEC 9798-2) の処理シーケンスを示す。図22においては、共通鍵暗号方式としてDESを用いているが、共通鍵暗号方式であれば他の方式も可能である。図22において、まず、Bが64ビットの乱数 R_b を生成し、 R_b および自己のIDであるID (b) をAに送信する。これを受信したAは、新たに64ビットの乱数 R_a を生成し、 R_a 、 R_b 、ID (b) の順に、DESのCBCモードで鍵 K_{ab} を用いてデータを暗号化し、Bに返送する。なお、鍵 K_{ab} は、AおよびBに共通の秘密鍵、認証鍵である。DESのCBCモードを用いた鍵 K_{ab} による暗号化処理は、例えばDESを用いた処理においては、初期値と R_a とを排他的論理和し、DES暗号化部において、鍵 K_{ab} を用いて暗号化し、暗号文 E_1 を生成し、続けて暗号文 E_1 と R_b とを排他的論理和し、DES暗号化部において、鍵 K_{ab} を用いて暗号化し、暗号文 E_2 を生成し、さらに、暗号文 E_2 とID (b) とを排他的論理和し、DES暗号化部において、鍵 K_{ab} を用いて暗号化して生成した暗号文 E_3 によって送信データ (Token-AB) を生成する。

【0140】これを受信したBは、受信データを、やはり共通の秘密鍵としてそれぞれの記録素子内に格納する鍵 K_{ab} (認証鍵) で復号化する。受信データの復号化方法は、まず、暗号文 E_1 を認証鍵 K_{ab} で復号化し、初期値と排他的論理和し乱数 R_a を得る。次に、暗号文 E_2 を認証鍵 K_{ab} で復号化し、その結果と E_1 を排他的論理和し、 R_b を得る。最後に、暗号文 E_3 を認証キー K_{ab} で復号化し、その結果と E_2 を排他的論理和し、ID (b) を得る。こうして得られた R_a 、 R_b 、ID (b) のうち、 R_b およびID (b) が、Bが送信したものと一致するか検証する。この検証に通った場合、BはAを正当なものとして認証する。

【0141】次にBは、認証後に使用するセッションキー (K_{ses}) を乱数によって生成する。そして、 R_b 、 R_a 、 K_{ses} の順に、DESのCBCモードで認証キー K_{ab} を用いて暗号化し、Aに返送する。

【0142】これを受信したAは、受信データを認証キー K_{ake} で復号化する。受信データの復号化方法は、Bの復号化処理と同様である。こうして得られた R_b 、 R_a 、 K_{ses} の内、 R_b および R_a が、Aが送信したものと一致するか検証する。この検証に通った場合、AはBを正当なものとして認証する。互いに相手を認証した後は、セッションキー K_{ses} は、認証後の秘密通信のための共通鍵として利用される。

【0143】なお、受信データの検証の際に、不正、不

一致が見つかった場合には、相互認証が失敗したものと
して、その後の相互間のデータ通信処理が禁止される。

【0144】図23、図24に本発明のデバイスとメディア間における相互認証、鍵（セッション鍵）共有処理フローを示す。図23、図24において、左側がデバイスのメモリインタフェース、右側がメディア2のコントローラにおける処理である。

【0145】まず、メディア2コントローラが乱数Raを生成（S401）し、Raおよび自己のIDであるメディア2IDをデバイスメモリインタフェースに送信（S402）する。これを受信（S403）したデバイスメモリインタフェースは、受信したメディア2IDと、初期値（IVake）の排他論理和に自己の所有する認証鍵生成用マスター鍵：MKakeを適用しDES暗号化処理を行なって認証鍵Kakeを生成（S404）する。さらに、デバイスメモリインタフェースは、新たに乱数Rbを生成（S405）し、初期値IVauthとRbとを排他的論理和し、鍵Kakeを用いて暗号化し、暗号文E1を生成し、続けて暗号文E1とRaとを排他的論理和し、鍵Kakeを用いて暗号化して暗号文E2を生成し、さらに、暗号文E2とメディア2IDとを排他的論理和し、鍵Kakeを用いて暗号化して暗号文E3を生成し（S406）、生成したデータE1||E2||E3をメディア2コントローラに送信（S407）する。[||]は、データの結合を意味する。

【0146】これを受信（S408）したメディア2コントローラは、受信データを、認証鍵Kakeで復号化（S409）する。受信データの復号化方法は、まず、暗号文E1を認証鍵Kakeで復号化し、初期値と排他的論理和し乱数Rb'を得る。次に、暗号文E2を認証鍵Kakeで復号し、その結果とE1を排他的論理和し、Ra'を得る。最後に、暗号文E3を認証鍵Kakeで復号し、その結果とE2を排他的論理和し、メディア2ID'を得る。こうして得られたRa'、Rb'、メディア2ID'のうち、Ra'およびメディア2ID'が、メディア2が送信したものと一致するか検証（S410、S411）する。この検証に通った場合、メディア2はデバイスを正当なものとして認証する。Ra'およびメディア2ID'が、送信データと不一致であったときは、相互認証が失敗（S413）したものと
し、その後のデータ通信を中止する。

【0147】次にメディア2コントローラは、認証後に使用するセッションキー（Kses）としての乱数を生成（S412）する。次に、図24のステップS421において、Ra、Rb、Ksesの順に、DESのCBCモードで認証鍵Kakeを用いて暗号化し、デバイスメモリインタフェースに送信（S422）する。

【0148】これを受信（S423）したデバイスメモリインタフェースは、受信データを認証鍵Kakeで復号（S424）する。こうして得られたRa''、R

b''、Ksesの内、Ra''およびRb''が、デバイスが送信したものと一致するか検証（S425、S426）する。この検証に通った場合、デバイスはメディア2を正当なものとして認証（S427）する。互いに相手を認証した後は、セッションキーKsesを共有（S429）し、認証後の秘密通信のための共通鍵として利用される。Ra''およびRb''が、送信データと不一致であったときは、相互認証が失敗（S428）したものと
し、その後のデータ通信を中止する。

【0149】図20に戻り、メディア2の認識処理について説明を続ける。ステップS305において上述の相互認証、鍵共有処理が実行され、ステップS306で相互認証が成功したことが確認されると、相互認証処理時に受信したメディア2のIDと、既にセットされているリポーションリスト中のリポーク（排除）メディア2のリストとの比較照合を実行（S307）する。

【0150】受信IDと一致するIDがリスト中に存在しなかった場合は、装着メディア2はリポーク対象メディアではなく、有効に利用可能なメディアであると判定（S308においてNo）し、ステータスレジスタのメディア2有効フラグを1（有効）にセット（S309）し、ビジーフラグを0（待機）にセット（S310）する。受信IDと一致するIDがリポーションリストにあった場合（S308においてYes）は、装着メディア2はリポーク対象メディアであり、有効に利用できないと判定し、ステップS309の有効フラグの有効化処理を実行せずステップS310でビジーフラグを0（待機）にセットして処理を終了する。

【0151】一方、制御部は、ステップS311において、ステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが0（待機）になったことを確認（S312）の後、メディアフラグ状態を確認して有効（フラグ：1）である場合（S313でYes）にのみ処理を続行し、無効（フラグ：0）である場合（S313でNo）は、処理を終了する。

【0152】次に、図21に進み、制御部は、メディア2に関するファイル割り当てテーブル呼び出しコマンドをメモリインタフェースに送信（S321）し、メモリインタフェースは、ファイル割り当てテーブルの格納されたセクタ読み出しコマンドをメディア2に送信（S322）し、ファイル割り当てテーブルをメディア2から受信し、制御部に送信（S323）する。

【0153】制御部は、メディア2に格納されたデータに対応するファイル割り当てテーブルを受信（S324）すると、テーブルに基づいてブロック・パーミッション・テーブル（BPT）の読み出し処理を実行（S325）し、BPTのセットコマンドと、BPTをメモリインタフェースに送信（S326）する。BPTのセット処理は、BPTが有効である場合にのみ実行され、BPTがセットされると、メディアからのコンテンツ書き

込み処理等、コンテンツ処理の際、BPTを参照してブロック毎の消去が可能か否かを判定する。実際のBPTを参照したデータ書き込み処理については、後段で説明する。

【0154】ブロック・パーミッション・テーブル(BPT)のセットコマンドと、BPTを制御部から受信(S327)すると、メモリインタフェースは、ステータスレジスタのビジーフラグを1(ビジー)にセット(S328)し、BPTの改竄チェック用の改竄チェック値(ICV)生成鍵Kicv_bptを生成(S329)する。

【0155】BPTの改竄チェック用の改竄チェック値(ICV)生成鍵Kicv_bptは、予めデバイス内に格納されたBPTのICV鍵を生成するマスター鍵:MKicv_bptと、BPTのICV鍵を生成する時の初期値:IVicv_bptと、メディア2IDに基づいて生成する。具体的には、改竄チェック値(ICV)生成鍵Kicv_bpt=DES(E, MKicv_bpt, データ2ID・IVicv_bpt)に基づいて生成される。式の意味は、メディア2IDと初期値(IVicv_bpt)の排他論理和にマスター鍵:MKicv_bptによるDESモードでの暗号化処理を実行するという意味である。

【0156】次にメモリインタフェースは生成した改竄チェック値(ICV)生成鍵Kicv_bptとIVbptを用いてBPTのICV'を生成し、予めBPT内に格納された正しいICV値との照合処理(ICV'=ICV?)を実行(S330)する。なお、ICV'の生成処理は、前述の図14で説明したDESモードに基づいて、初期値IVbptを用い、生成した改竄チェック値

(ICV)生成鍵Kicv_bptを適用した処理によって行われる。なお、BPTの付帯情報として格納されたICVは、メディア2IDを含むデータに基づいて生成されており、ICVのチェックは、BPTのデータ改竄の有無のみならず、メディア固有の正当なBPT、すなわち他のメディアにコピーされたBPTでないことの検証も兼ね備える機能を持つ。

【0157】ICV'=ICVである場合(S331でYes)は、BPTが正当なメディアに格納された改竄のない正当なものであると判定され、コンテンツ処理の際に参照可能な状態にセットし、メディア2セットフラグを1(セット)にセット(S332)する。ICV'≠ICVである場合(S331でNo)は、BPTに改竄ありと判定され、BPTの参照処理に基づくコンテンツ処理を禁止し処理を終了する。以上の処理の終了により、ビジーフラグは0にセット(S333)される。

【0158】一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信(S334)し、ビジーフラグが0となったことを条件(S335でYes)としてメディア2セットフラグを保存(S336)

する。保存されるメディア2セットフラグは、BPTの改竄が無いと判定された場合は、メディア2が有効にセットされたことを示す1、その他の場合は0となる。

【0159】(データファイル読み出し処理)次に、データファイルの読み出し処理について図25のフローを用いて説明する。データファイルには、音楽データ、画像データ等のコンテンツデータファイル、さらに前述したリボケーションリストも含まれる。図25に示すフローは、内部メモリ、外部メモリ(メディア1、メディア2)のいずれかに格納されたデータファイルの読み出しに共通な処理フローである。図25において、左側がデバイスの制御部、右側がデバイスのメモリインタフェースの処理である。

【0160】まず、制御部は、ファイル割り当てテーブル(図17参照)から読み出し対象データのセクタアドレス(S(1)~S(k))を取得(S501)し、メモリインタフェースに取得したセクタS(i)読み出しコマンドを順次送信(S502, S503)する。メモリインタフェースは、セクタS(i)読み出しコマンドを受信(S504)すると、ビジーフラグを1(ビジー)に設定(S505)し、受信セクタS(i)が内部メモリか、外部メモリであるかを判定(S506)し、外部メモリである場合は、メディア1かメディア2のセットフラグが1(メディアが有効にセットされていることを示す)であるかを判定(S507)し、セットフラグが1である場合には、さらにブロックパーミッション・テーブル(BPT)を参照して、BPTが読み出し対象であるセクタS(i)を読み出し許可対象ブロックとして設定しているかを判定(S508)する。BPTに読み出し許可ブロックの設定がある場合には、外部メモリから該当セクタのデータを読み出す(S509)。

【0161】なお、読み出し対象データがBPTによる管理のなされていない内部メモリ内のデータである場合は、ステップS507, S508はスキップする。ステップS507, S508の判定がNoである場合、すなわちセクタS(i)を格納したメディアのセットフラグが1でない場合、または、BPTにセクタS(i)の読み出し許可が設定されていない場合には、ステップS513に進み、読み出しエラーとして読み出し成功フラグが0にセットされる。

【0162】ステップS506~S508の判定ブロックにおいて、対象セクタS(i)の読み出しが実行可と判定されると、メモリから該当セクタが読み出され、セクタに対応して設定されている冗長部の誤り訂正符号に基づく誤り訂正処理が実行(S510)され、誤り訂正が成功した(S511)ことを確認し、読み出し成功フラグを1(成功)にセットし、読み出し結果をバッファに格納(S512)し、ビジーフラグを0(待機)に設定(S513)する。誤り訂正に失敗した場合は、読み出し成功フラグを0(失敗)に設定(S513)して処

理を終了する。

【0163】また、制御部は、ステップS515～S520において、メモリインタフェースのステータスを読み出して、ビジーフラグが0の状態において、読み出し成功フラグが1であることを条件として読み出しデータをバッファから取り出して保存し、アドレスを順次インクリメントして、データを順次バッファから取り出して保存する処理を繰り返し実行し、すべての読み出し対象セクタを保存した後、全読み出しセクタデータからファイルを構成して処理を終了する。

【0164】（ファイル書き込み処理）次に、データファイルの書き込み処理について図26のフローを用いて説明する。図26に示すフローは、内部メモリ、外部メモリ（メディア1、メディア2）のいずれかにファイルを書き込む際の共通処理フローである。図26において、左側がデバイスの制御部、右側がデバイスのメモリインタフェースの処理である。

【0165】まず、制御部は、書き込み対象ファイルをセクタに分割する。分割されたデータをD(1)～D(k)とする。制御部は、次に各データD(i)の書き込みセクタS(i)を設定して、メモリインタフェースにセクタS(i)書き込みコマンドと、データD(i)を順次送信（S602～S604）する。メモリインタフェースは、セクタS(i)書き込みコマンドを受信（S605）すると、ビジーフラグを1（ビジー）に設定（S606）し、受信セクタS(i)が内部メモリか、外部メモリであるかを判定（S607）し、外部メモリである場合は、メディア1かメディア2のセットフラグが1（メディアが有効にセットされていることを示す）であるかを判定（S608）し、セットフラグが1である場合には、さらにブロックパーミッション・テーブル（BPT）を参照して、BPTが書き込み対象であるセクタS(i)を書き込み許可対象ブロックとして設定しているかを判定（S609）する。BPTに書き込み許可ブロックの設定がある場合には、セクタに対応して設定する誤り訂正符号を生成（S610）し、セクタS(i)にデータD(i)と誤り訂正符号を持つ冗長部を書き込み、書き込み成功フラグを1（成功）にセットし、ビジーフラグを0（待機）に設定（S614）する。

【0166】なお、書き込み対象データがBPTによる管理のなされていない内部メモリ内への書き込み処理である場合は、ステップS608、S609はスキップする。ステップS608、S609の判定がNoである場合、すなわちメディアのセットフラグが1でない場合、または、BPTにセクタS(i)の書き込み許可が設定されていない場合には、ステップS613に進み、書き込みエラーとして書き込み成功フラグを0にセットする。

【0167】また、制御部は、ステップS616～S6

20において、メモリインタフェースのステータスを読み出して、ビジーフラグが0の状態において、書き込み成功フラグが1であることを条件としてアドレスを順次インクリメントして、書き込みデータを順次メモリインタフェースに送信する。すべての処理が終了すると、ファイル割り当てテーブルの更新処理を実行（S621）し、更新したファイル割り当てテーブルを更新コマンドとともにメモリインタフェースに送信（S622）し、メモリインタフェースはコマンドに従ってファイル割り当てテーブルの書き込み処理を実行（S623）する。

【0168】〔セクタ位置に応じた暗号化鍵を適用した暗号化処理〕次に、セクタ位置に応じた暗号化鍵を適用した暗号化処理について説明する。著作権などを保護するためにコンテンツ部に対する暗号化を行う場合があるが、コンテンツ部全体に対して一つの暗号化鍵を使って暗号化すると、同一の鍵の元での大量の暗号文が発生し、攻撃が容易となってしまう危険性がある。通常はコンテンツ部をできるだけ分割し、それぞれを異なる鍵で暗号化の方が望ましいと言える。本システムでのコンテンツ暗号化の最小単位として、セクタが挙げられるが、ヘッダ領域に鍵を保存するという目的の場合には、セクタの数だけ8バイト（DESの場合）または16バイト（トリプルDES（Triple-DESの場合））の鍵情報が必要となるためヘッダのサイズが膨大になってしまい限られたメモリ領域のデータ領域を減少させてしまうことになり、実用上好ましくない。また、各セクタのデータ部分にそのセクタを暗号化するための鍵を格納する方法をとればヘッダサイズに影響を及ぼすことはないが、鍵の領域にはデータを置けなくなるためデータサイズが目減りしてしまうことと、万一、制御部側でファイルシステムを持つようなシステムの場合にはファイルシステム自体に大幅な変更を必要とする。

【0169】そこで、本発明のシステムでは、先に説明した各コンテンツの属性情報であるセキュリティヘッダ（図7参照）の中に例えば、メディアの1ブロックあたりのセクタ数Mに対応するM個の鍵情報を格納し、これらを各セクタに対する暗号化鍵として適用する（図8参照）。図7に示したセキュリティヘッダ中のKc_Encrypted0～Kc_Encrypted31が32個の暗号化鍵Kcを示す。なお、[Encrypted]は、それぞれの鍵Kcが暗号化されて格納されていることを示す。これらの複数の鍵の中からセクタのブロック内位置によって鍵を選択してセクタ対応の暗号化鍵として用いる構成とした。

【0170】図27に、コンテンツのヘッダ情報としてコンテンツに対応して生成されるセキュリティヘッダにおける鍵格納構成と、各格納鍵と、各鍵の適用対象となるメモリ内の各セクタとの対応を説明する図を示す。図27(a)が先に図7を用いて説明したセキュリティヘッダ内の鍵格納構成を簡略化して示した図である。図27(a)のセキュリティヘッダには、Kc(0)～Kc

(M-1) までのM個の鍵(コンテンツキー)が格納されている。ヘッダには鍵以外にもバージョン、コンテンツタイプ等の各種情報が格納され、さらにヘッダ情報の改竄チェック用のICVが格納されている。

【0171】このM個のコンテンツキーは、例えば図27(b)に示すように各々が各セクタに対応付けられて各セクタに格納するデータの暗号化に使用される。先に図3を用いて説明したように、ブロック単位での消去を行なうフラッシュメモリは、図27(b)に示すようにデータ格納領域がブロック単位に分割され、各ブロックはさらに複数セクタに分割されている。例えば鍵Kc(0)を、メモリの各ブロックのセクタ0に格納するデータの暗号化鍵として適用し、鍵Kc(s)を、メモリの各ブロックのセクタsに格納するデータの暗号化鍵とする。さらに、鍵Kc(M-1)を、メモリの各ブロックのセクタM-1に格納するデータの暗号化鍵として適用する。

【0172】このように、セクタに対応して異なる暗号鍵を適用してデータを格納することにより格納データ(ex. コンテンツ)のセキュリティが高められる。すなわち、コンテンツ全体を1つの鍵で暗号化した場合、鍵漏洩によるコンテンツ全体の復号が可能となるのに対し、本構成によれば、1つの鍵の漏洩によってデータ全体を復号することは不可能であるからである。

【0173】暗号化アルゴリズムは、例えば1つの暗号鍵によるDES暗号化処理を実行するシングルDESが適用される。また、シングルDESではなく、暗号化に2つ以上の鍵を使用するトリプルDES(Triple DES)を適用した暗号化構成としてもよい。

【0174】トリプルDES(Triple DES)の詳細構成例を図28に示す。図28(a)、(b)に示すようにトリプルDES(Triple DES)としての構成には、代表的には以下のような2つの異なる態様がある。図28(a)は、2つの暗号鍵を用いた例を示すものであり、鍵1による暗号化処理、鍵2による復号化処理、さらに鍵1による暗号化処理の順に処理を行う。鍵は、K1、K2、K1の順に2種類用いる。図28(b)は3つの暗号鍵を用いた例を示すものであり、鍵1による暗号化処理、鍵2による暗号化処理、さらに鍵3による暗号化処理の順に処理を行い3回とも暗号化処理を行う。鍵は、K1、K2、K3の順に3種類の鍵を用いる。このように複数の処理を連続させる構成とすることで、シングルDESに比較してセキュリティ強度を向上させることが可能である。

【0175】図29に、メモリに格納するデータの各セクタ毎に異なる2つの暗号鍵のペアを適用してトリプルDESによる暗号化処理を行なった構成例を示す。図29に示すように、各ブロックのセクタ0は、鍵Kc(0)とKc(1)の2つの鍵を用いてトリプルDES暗号化を行ない、セクタsは、鍵Kc(s)とKc(s

+1)の2つの鍵を用いてトリプルDES暗号化を行ない、セクタM-1は、鍵Kc(M-1)とKc(0)の2つの鍵を用いてトリプルDES暗号化を行なう。この場合でも、ヘッダに格納する鍵数は、M個であり、図27(a)で示した鍵格納数を増加させる必要はなく、セキュリティを高めることが可能となる。

【0176】さらに、図30に異なる態様でのデータ暗号化構成例を示す。図30は、メモリの各ブロックの2つの連続するセクタ領域を1つの暗号化ブロックとして、2つの鍵を用いてトリプルDES暗号化を行なった態様である。図30に示すように、各ブロックのセクタ0とセクタ1は、鍵Kc(0)とKc(1)の2つの鍵を用いてトリプルDES暗号化を行ない、セクタ2sとセクタ2s+1は、鍵Kc(2s)とKc(2s+1)の2つの鍵を用いてトリプルDES暗号化を行ない、セクタM-2とセクタM-1は、鍵Kc(M-2)とKc(M-1)の2つの鍵を用いてトリプルDES暗号化を行なう。このように複数のセクタに同一の暗号化処理を適用することで暗号化プロセスまたは復号プロセスの処理軽減を可能とすることができる。

【0177】図27、図29、図30に示す例の他にも、ヘッダに複数鍵を格納し、その複数鍵から選択した鍵を用いてセクタ毎の暗号化を実行する構成としては様々な構成が可能である。例えば、図27、29、30では、セクタ数と同数の鍵をヘッダに格納する構成としているが、例えばセクタ数がMのとき、格納鍵数をN(N<M)として、セクタ0とセクタsは同じ鍵で暗号化する等の構成としてもよい。また格納鍵数をL(L>M)として、各セクタごとに全く異なる複数の鍵セットによるトリプルDESを適用する構成としてもよい。

【0178】[セクタ単位の改竄チェック値(ICV)の付加構成]次に、セクタ単位の改竄チェック値(ICV)の付加構成について説明する。複数セクタにまたがって構成されるデータについて、その正当性を確認する場合、一般には、コンテンツデータ全体の最後などに前述した改竄チェック値(ICV)を付加させる構成とするのが一般的であった。このようなデータ全体のICVの付加構成においては、データを構成している各セクタ単位で、正当性を確認することができない。

【0179】またICVを格納する場合、実データであるコンテンツの格納領域と同領域にICVを入れ込むと、その分データ部として使用できる領域が減ってしまう。もし、各セクタにセクタ内のデータに対してセクタ毎のICVを入れ込むと、デバイスのファイルシステムはデータ部単位でデータを読み出す処理を実行するために、実際に使用されるデータをICVから切り離して取り出すための処理、すなわち一度、読み出したデータ部内のセクタ内のICVを取り除く処理と、取り出したセクタ内のデータを複数セクタで連結する処理を実行することが必要となり、その処理を実行するためのファイル

システムを新たに構築することが必要となる。さらに、これらの I C V チェックを制御部で行うとなると、制御部にその処理の分の負荷がかかってしまう。

【0180】本発明のデータ処理装置においては、セクタ毎にデータ改竄チェックを可能とするため、セクタ毎に I C V を設定し、その I C V 設定位置を実データ領域ではなく、デバイスのファイルシステムによって読み取られない領域として予め設定されている冗長部領域とした。冗長部に I C V を置く構成とすることで、データ内に I C V を置く必要がなくなり、データ部の領域が多く利用できる。また、冗長部に I C V を置くことで、データ部と I C V の切り分け・データ連結処理が不要となるために、データ読み出しの連続性が保たれる。

【0181】データを読み出す時には、メモリインタフェース部 300 (図 2 参照) でセクタ毎の I C V チェック処理を実行し、改竄ありと判定され無効なデータである場合は制御部 205 (図 2 参照) への転送を実行しない。また、データ書き込み時には、メモリインタフェース部 300 において各セクタの I C V を計算して、冗長部に書きこむ処理を実行する。

【0182】なお、各セクタで I C V を付加するかしないかを、セキュリティヘッダ (Security Header) に記述して指定する。この構成については、図 7 のセキュリティヘッダ構成の説明中に示したように、セキュリティヘッダ中の I C V フラグ (ICV Flag) が、ブロック内のセクタ数 (32 セクタ) 分のフラグを持ち、ブロック内の各セクタの I C V 付加・非付加を示す。例えば 0 : I C V なし、1 : I C V あり、として設定される。

【0183】各セクタのデータ利用部と冗長部構成を図 31 に示す。図 31 (a) のように、メモリ (フラッシュメモリ) に格納されるデータは複数のセクタ領域を持つブロック単位領域に分割して格納される。(b) に示すように、各セクタはデバイスのファイルシステムによって実データ (e x. コンテンツ) として読み取られる例えば 512 あるいは 1024 バイトのデータ利用部と、ファイルシステムによっては読み取られない E C C (Error Correction Code) 等の情報を格納した冗長部とによって構成される。

【0184】この冗長部の容量は例えば 16 バイト、あるいは 20 バイトの予め決められた領域であり、デバイスのファイルシステムは、この冗長部を非データ領域として認識し、データ (コンテンツ) 読み取り処理においては読み取らない。一般に、冗長部に格納される E C C は、冗長部全体を使用せず、冗長部には非使用領域 (リザーブ領域) が存在する。このリザーブ領域に各セクタの改竄チェック値 (I C V) を格納する。

【0185】冗長部に I C V を格納した場合のデバイスのファイルシステムによるデータ部の連結処理は、図 31 (c) に示すように、純粹にデータとして使用するものだけが格納されたデータ部の連結を行なうのみの従来

のデータ連結処理と同様の処理が可能となる。従って、デバイスのファイルシステムは、冗長部を除くデータ部領域を単に連結すればよく、新たな処理は何ら必要としない。

【0186】本構成により、複数のセクタで構成されるデータのセクタ単位でデータの正当性の検証することが出来る。また、改竄チェック用の I C V を冗長部に入れることで、データ用に使えるデータ領域をそのまま活用することが出来る。また、制御部には、I C V チェックの結果、正しい (改竄なし) と判定された正しいセクタのみが送信される。また、I C V チェックがメモリインタフェース部にて行われるので、制御部の負担がかからない等の効果がある。

【0187】[メディア内の個別鍵によるコンテンツ鍵の保存処理] 次に、メディア内の個別鍵によるコンテンツ鍵の保存処理構成について説明する。先に、図 7 を用いて説明したように、コンテンツに対応して構成されるセキュリティヘッダには、セクタ対応の暗号鍵としての複数のコンテンツキー (K c _Encryptedxx)、およびコンテンツチェック値生成鍵 (K i c v _Encrypted) が暗号化されて格納されている。

【0188】これらの鍵の暗号化の 1 つの態様は、予めデバイスのメモリインタフェースのメモリ部 321 (図 4 参照) に格納されている配送鍵 K d i s t によって暗号化して格納する構成がある。例えば、K c _Encrypted0 = E n c (K d i s t, K c (0)) である。ここで、E n c (a, b) は、b を a で暗号化したデータであることを示す。このように、それぞれの鍵をデバイスの配送鍵 K d i s t を用いて暗号化してセキュリティヘッダに格納する構成が 1 つの構成である。

【0189】さらに、メディア 2、すなわち暗号処理部を持ち、デバイスとの相互認証を実行してコンテンツ処理を実行するメディアにおいて、メディア 2 の固有鍵を用いてメディア 2 に格納するコンテンツに関するコンテンツキー、I C V 生成鍵を暗号化する態様がある。以下、メディア 2 の固有鍵、ここではメディア 2 保存鍵 K s t o を用いて暗号化したコンテンツキー、コンテンツ I C V 生成鍵をセキュリティヘッダに格納する処理について説明する。

【0190】メディア 2 保存鍵 K s t o は、図 2 に示したようにメディア 2、230 のメディア 2 コントローラ 231 の内部メモリ 235 に格納されている。従って、メディア 2 保存鍵 K s t o を使用したコンテンツキー、I C V 生成鍵の暗号化処理、復号処理はメディア 2 側で実行される。メディア 2 を装着したデバイスが、メディア 2 のコンテンツ利用に際し、コンテンツキー、I C V 生成鍵を取得、あるいはセキュリティヘッダへの格納処理を実行する場合は、メディア 2 側で鍵の暗号化、復号処理を実行することが必要となる。本発明のデータ処理装置においては、これらを C B C (Cipher Block Chain

ing) モードで処理することを可能とした。

【0191】図32にCBCモードにおける鍵の暗号化処理構成を示す。この暗号化処理は、メディア2の暗号処理部236(図2参照)において実行される。内部メモリ235に格納された初期値IV_keysと、コンテンツチェック値生成鍵Kicv_contとの排他論理和を実行し、その結果をメディア2の内部メモリ235に格納された保存鍵Kstoを適用したDES暗号化を行ない、その結果をKicv_cont Encryptedとしてヘッダに格納する。さらに、Kicv_cont Encryptedと、セクタ(0)に対応するセクタ対応コンテンツキーKc(0)との排他論理和を実行し、その結果をメディア2の内部メモリ235に格納された保存鍵Kstoを適用したDES暗号化を行ない、その結果をKc(0) Encryptedとしてヘッダに格納する1つの暗号化コンテンツキーとする。さらに、Kc(0) Encryptedと、セクタ(1)に対応するセクタ対応コンテンツキーKc(1)との排他論理和を実行し、その結果に対して保存鍵Kstoを適用したDES暗号化を行ない、その結果をKc(1) Encryptedとする。以下、これらの処理を繰り返し実行して、ヘッダ格納用の鍵データとする。

【0192】次に、図33にCBCモードにおける鍵の復号処理構成を示す。この復号処理は、メディア2の暗号処理部236(図2参照)において実行される。まず、Kc(0) Encryptedに対して、メディア2の内部メモリ235に格納された保存鍵Kstoを適用したDES復号処理を行ない、その結果を内部メモリ235に格納された初期値IV_keysと排他論理和することにより、セクタ(0)に対応するセクタ対応コンテンツキーKc(0)が出力される。さらに、Kc(1) Encryptedに対して、保存鍵Kstoを適用したDES復号処理を行ない、その結果をコンテンツキーKc(0) Encryptedと排他論理和することにより、セクタ(1)に対応するセクタ対応コンテンツキーKc(1)が出力される。以下、これらの処理を繰り返し実行して、コンテンツキーを取得する。なお、図には、コンテンツキーのみを出力データとした例を示しているが、コンテンツ改竄チェック値生成鍵(Kicv_Encrypted)についても同様の処理が適用可能であり、暗号化されたコンテンツ改竄チェック値生成鍵(Kicv_Encrypted)からコンテンツ改竄チェック値生成鍵(Kicv)の生成が可能である。

【0193】上述のセクタ対応コンテンツキーKc(x)またはコンテンツ改竄チェック値生成鍵(Kicv)の暗号化、復号処理は、多くの場合、メディア2を装着したデバイスからのコマンドに基づいて実行される。この場合、デバイスとメディア2間では前述した相互認証が実行され、相互認証処理が成立したことを条件としてコンテンツ再生、格納等の様々な処理が実行され、その一連のコンテンツ処理の1つとして上述のコン

テンツキーの復号、暗号化処理が実行されることになる。復号された鍵(ex. コンテンツキーKc(x))をデバイスとメディア2間において転送する場合は、相互認証時に生成したセッションキーKsesで暗号化される。このセッションキーKsesによる暗号化、復号処理もCBCモードを適用することで、よりセキュリティを高めることが可能となる。

【0194】図34にメディア2において、セキュリティヘッダに格納された鍵をDES-CBCモードで復号し、復号した鍵データをさらにセッションキーKsesを適用してDES-CBCモードで暗号化する処理構成を示す。図34の上段は、図33と同様の構成であり、セキュリティヘッダから取り出した暗号化されたコンテンツキーを順次DES復号部に入力してメディア2の保存鍵Kstoを適用して復号処理を実行し、出力結果を初期値、または入力データ列の前データと排他論理和して、出力結果としてのコンテンツキーを取得する。

【0195】これらの出力された結果をさらに、デバイスとの相互認証時に生成したセッションキーKsesを適用したDES-CBCモードでの暗号化処理を実行する。その結果得られたSE0~SEM-1:Kc(0) Encrypted~Kc(M-1) Encryptedをデバイスに送信する。デバイス側では、受信したデータ列Kc(0) Encrypted~Kc(M-1) Encryptedについて、メディア2との相互認証時に生成したセッションキーKsesを適用して、図33と同様のDES-CBCモードでの復号処理を実行することによりコンテンツキーK(c)を取得することができる。なお、図には、コンテンツキーのみを処理データとした例を示しているが、コンテンツ改竄チェック値生成鍵(Kicv_Encrypted)についても同様に処理データとして構成することが可能である。

【0196】[暗号化データの読み出し処理] 図35以下のフローを用いて、暗号化されたデータのメディアからの読み出し処理の詳細を説明する。なお、データの暗号化態様は、上述したようにセクタ毎に異なる鍵で暗号化した態様と、コンテンツ全体を1つの暗号化鍵で暗号化した態様とがあり、これらは、ヘッダの情報に基づいて判定される。図35のフローにおいて左側はデバイスの制御部、右側はデバイスのメモリインタフェースの処理である。

【0197】まず制御部は、読み出し対象となるコンテンツのヘッダファイルを読み出す(S701)。この処理は、前述の図25のファイル読み出し処理フローに従った処理として実行される。次にヘッダセットコマンドと、読み出したヘッダファイルをメモリインタフェースに送信(S702)する。

【0198】メモリインタフェースはヘッダセットコマンドを受信(S703)すると、ビジーフラグを1(ビジ)にセット(S704)し、ヘッダの改竄チェック

値 (ICV) を検証 (S705) する。ヘッダの ICV チェックは、先に図 14 を用いて説明した ICV 生成処理において、セキュリティヘッダ検証値生成鍵 K_{icv_sh} と、初期値 IV_{sh} を適用してヘッダの構成データを入力して ICV' を生成し、生成した ICV' と予めヘッダに格納された ICV とを照合する処理によって実行する。

【0199】検証によりヘッダ改竄なしと判定 (S706) されると、ヘッダ内の有効リポケーションリスト・バージョンが 0 でないかがチェック (S707) され、例えば、自デバイスで生成し格納したコンテンツをメモリに格納するとき等は、リポケーションリスト・バージョンを 0 として、再生処理等の際にリポケーションリストを非参照とした処理を実行可能とする。

【0200】リポケーションリスト・バージョンが 0 の場合は、リポケーションリストを参照する必要がないのでステップ S710 に進む。バージョンが非 0 であるときは、現在セットされているリポケーションリストが、ヘッダのバージョンより古くないかをチェック (S708) し、古い場合は、S713 に進み、ヘッダセット成功フラグを 0 (NG) に設定して処理を終了する。セットされているリポケーションリストがヘッダのバージョンより古くなければ、ステップ S709 に進み、リポケーションリストを参照して、読み出し対象のコンテンツ ID がないかを判定する。あった場合は読み出しを禁止する処理として、ステップ S713 でヘッダセット成功フラグを 0 (NG) として処理を終了する。

【0201】リポケーションリストに読み出し対象コンテンツ ID が記録されていない場合は、ステップ S710 進み、ヘッダ情報に基づいて暗号化されたコンテンツキー K_c と、コンテンツチェック値生成鍵 K_{icv_cont} を復号する。なお、リポケーションリストは、先の図 16 の起動時フローにおいて説明したように、起動時にメモリインタフェースにセットアップされ、セットアップ後は、メディアの装着時、コンテンツの再生時においてメモリインタフェースにおいて継続的に利用可能としたリポケーションリストである。

【0202】先に、図 7 他を用いて説明したようにセキュリティヘッダの中には、前述のセクタ毎に適用する暗号鍵としての複数のコンテンツキー $K_c(0) \sim K_c(M-1)$ が暗号化されて格納されている。また、コンテンツの改竄チェック値 (ICV) を生成するためのコンテンツチェック値生成鍵 K_{icv_cont} も暗号化されて格納されている。

【0203】コンテンツの復号に先立ち、これらのコンテンツチェック値生成鍵 K_{icv_cont} を復号してコンテンツの改竄チェックを実行する処理が必要であり、また、コンテンツキー $K_c(0) \sim K_c(M-1)$ を復号する処理が必要となる。

【0204】図 37 に暗号化されたコンテンツキー K

c、コンテンツチェック値生成鍵 K_{icv_cont} の復号処理フローを示す。図 37 の各ステップについて説明する。図 37 の処理は、デバイスのメモリインタフェースにおける処理である。図 4 の暗号処理部 320 において実行される。

【0205】まず、暗号化コンテンツチェック値生成鍵 K_{icv_cont} を復号対象として選定 (S801) し、次に、ヘッダの暗号化フォーマットタイプ・フィールドの設定が 0 か否かを判定 (S802) する。暗号化フォーマットが 0 である場合は、コンテンツ全体をセクタに係わらず 1 つの暗号化態様としたデータ構成であり、暗号化フォーマットタイプ・フィールドの設定が 1 である場合は、前述の図 27 他で説明したセクタ単位の暗号化鍵を用いた方法である。セクタ単位の暗号化鍵を用いた方法である場合は、ステップ S803 に進み、セクタ毎に設定された暗号化コンテンツキー ($K_{c_Encrypted0} \sim 31$) を復号対象にする。

【0206】ステップ S802 で暗号化フォーマットが 0 であると判定された場合は、ステップ S804 でさらに、ヘッダの暗号化アルゴリズムフィールドをチェックして 1 (トリプル DES) が 0 (シングル DES) であるかを判定する。シングル DES である場合は、ステップ S805 で 1 つの暗号化コンテンツキー ($K_{c_Encrypted0}$) のみを復号対象として加え、トリプル DES である場合は、ステップ S806 で複数の暗号化コンテンツキー ($K_{c_Encrypted0}, 1$) を復号対象として加える。

【0207】次に、ステップ S807 において、ヘッダのコンテンツタイプフィールドの設定をチェックし、設定が 2 または 3 (メディア 2 の格納コンテンツ) でない場合は、ステップ S808 で、メモリ部 321 (図 4 参照) に格納された配送鍵 K_{dist} で復号対象データ、すなわち、暗号化コンテンツチェック値生成鍵 K_{icv_cont} と、1 以上のコンテンツキーを復号する。

【0208】設定が 2 または 3 (メディア 2 の格納コンテンツ) である場合は、ステップ S809 で復号対象データ、すなわち、暗号化コンテンツチェック値生成鍵 K_{icv_cont} と、1 以上のコンテンツキーをメディア 2 の保存鍵 K_{sto} (CBC モード) で復号する。この復号処理の詳細は、図 32、図 33、図 34 を用いて説明した通りである。

【0209】ステップ S809 におけるメディア 2 の保存鍵による暗号化コンテンツチェック値生成鍵 K_{icv_cont} と、1 以上のコンテンツキー K_c の復号処理について図 38 のフローを用いて説明する。図 38 のフローは、左側にデバイスのメモリインタフェース、右側にメディア 2 のコントローラ (図 2 参照) の処理を示している。

【0210】まず、メモリインタフェースは、復号対象データ $K(0) \sim K(n-1)$ (暗号化コンテンツチェ

10

20

30

40

50

ック値生成鍵 K_{icv_cont} と、1以上のコンテンツキーを設定 (S1001) し、CBC復号初期化コマンドをメディア2コントローラに送信 (S1003) し、メディア2コントローラは IV_{Keys} をレジスタにセット (S1005) する。その後、メモリインタフェースは、各鍵を順次送信 (S1004) し、メディア2コントローラが復号対象データ $K(i)$ を受信 (S1005) する。

【0211】次にメディア2コントローラは、受信した復号対象データ $K(i)$ に対して、メディア2の保存鍵 K_{sto} を用いたCBCモードによる復号処理を実行 (S1007) し、復号された鍵データ (ex. 複数のセクタ対応コンテンツキー) を取得 (S1008) する。次に、メディア2コントローラは、復号鍵データ列を、デバイスとの相互認証時に生成したセッションキーによってCBCモードでの暗号化処理を実行し、データ列 $K'(i)$ を生成して、結果をデバイスに送信 (S1009) する。ステップS1007～S1009の処理は、先に説明した図34のDES-CBCモードによる処理に基づいて実行される。

【0212】デバイスのメモリインタフェースは、順次 $K'(i)$ を受信し、すべてのデータを受信したことを確認の後、CBC終了コマンドをメディア2コントローラに送信する。メディア2コントローラはCBC終了コマンドの受信によりレジスタをクリア (S1014) する。

【0213】デバイスのメモリインタフェースは、メモリ部321 (図4参照) に格納した初期値 IV_{keys} を用い、メディア2との相互認証時に生成したセッションキー K_{ses} を適用してCBCモードでメディア2から受信した $K'(i)$ を復号 (S1010～S1013, S1015) する。この復号処理は、先に説明した図33の構成と同様の処理である。

【0214】上記処理により、デバイスは、ヘッダに格納された暗号化されたコンテンツキー K_c 、コンテンツチェック値生成鍵 K_{icv_cont} を復号し、それぞれの鍵を取得することができる。

【0215】次に図35に戻り、暗号化ファイルの読み出し処理の続きを説明する。上記の鍵復号処理ステップであるステップS710を終了すると、ステップS711に進む。ステップS711では、デバイスのメモリインタフェースはヘッダを「読み出しヘッダ」として内部に設定し、ヘッダセット成功フラグを1 (成功) にセットし、ビジーフラグを0 (待機) (S714) 設定する。コンテンツ読み出しに際しては、設定されたヘッダの情報に基づく処理が実行される。

【0216】一方、制御部側は、ステップS715でステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが0 (待機) (S716) であり、ヘッダセット成功フラグが1 (成功) (S717) とな

ったことを条件として次の処理 (図36) に進む。

【0217】図36のステップS721において、制御部は、ファイル割り当てテーブルから読み出し対象のコンテンツファイルのセクタアドレス (S(1)～S(k)) を取得し、メモリインタフェースに対して順次、セクタ $S(i)$ の読み出しコマンドを送信する。

【0218】メモリインタフェースは、セクタ $S(i)$ の読み出しコマンドを受信 (S724) すると、ビジーフラグを1 (ビジー) に設定 (S725) し、ヘッダ成功フラグが1 (成功) であることを条件 (S726) として次ステップに移行する。ヘッダ成功フラグが1 (成功) でない場合は、ステップS738に進み、読み出し成功フラグを0 (NG) として処理を終了する。

【0219】ヘッダ成功フラグが1 (成功) である場合は、受信セクタ $S(i)$ が内部メモリか、外部メモリであるかを判定 (S727) し、外部メモリである場合は、メディア1かメディア2のセットフラグが1 (メディアが有効にセットされていることを示す) であるかを判定 (S728) し、セットフラグが1である場合には、さらにブロックパーミッション・テーブル (BPT) を参照して、BPTが読み出し対象であるセクタ $S(i)$ を読み出し許可対象ブロックとして設定しているかを判定 (S729) する。BPTに読み出し許可ブロックの設定がある場合には、外部メモリから該当セクタのデータを読み出す (S730)。

【0220】なお、読み出し対象データがBPTによる管理のなされていない内部メモリ内のデータである場合は、ステップS728、S729はスキップする。ステップS728、S729の判定がNoである場合、すなわちセクタ $S(i)$ を格納したメディアのセットフラグが1でない場合、または、BPTにセクタ $S(i)$ の読み出し許可が設定されていない場合には、ステップS738に進み、読み出しエラーとして読み出し成功フラグが0にセットされる。

【0221】ステップS726～S729の判定ブロックにおいて、対象セクタ $S(i)$ の読み出しが実行可と判定されると、メモリから該当セクタが読み出され、セクタに対応して設定されている冗長部の誤り訂正符号に基づく誤り訂正処理が実行 (S731) され、誤り訂正が成功した (S732) ことを確認する。次に、ヘッダのICVフラグ (図7参照) を参照し、読み出し対象セクタが改竄チェック値 (ICV) による処理対象であるかを判定する。先に図31を用いて説明したように各セクタは、その冗長部に改竄チェック用のICVを格納しており、セクタ単位での改竄チェックが可能である。

【0222】ICVによる改竄チェックの対象である場合は、ステップS734において、ステップS710の復号処理によって得たコンテンツチェック値生成鍵 K_{icv_cont} と、初期値 IV_{cont} を適用し改竄チェック対象データ (セクタデータ) を入力して図14を用い

て説明した I C V 生成処理を実行し、I C V' を求め、セクタの冗長部に格納されている I C V との照合を行ない一致していれば改竄なしと判定する。

【0223】 I C V チェックにより改竄なしと判定されると、ステップ S 7 3 7 に進み、ヘッダ情報に基づいてデータの復号処理を実行して読み出し成功フラグを 1 (成功) に設定し、復号データをバッファに格納する。

【0224】 また、制御部は、ステップ S 7 4 0 ~ S 7 4 6 において、メモリインタフェースのステータスを読み出して、ビジーフラグが 0 の状態において、読み出し成功フラグが 1 であることを条件として読み出しデータをバッファから取り出して保存し、アドレスを順次インクリメントして、データを順次バッファから取り出して保存する処理を繰り返し実行し、すべての読み出し対象セクタを保存した後、全読み出しセクタデータからファイルを構成して処理を終了する。

【0225】 図 3 6 のステップ S 7 3 6 のデータ部復号処理の詳細を図 3 9 を用いて説明する。この復号処理はデバイスのメモリインタフェースの暗号処理部 3 2 0 (図 4 参照) において実行される。

【0226】 まず、復号対象のデータ格納セクタ位置を s ($0 \leq s \leq 31$ (セクタ数 32 の場合)) とする (S 1 1 0 1)。次にそのセクタが暗号化対象であるかをチェック (S 1 1 0 2) する。このチェックは、セキュリティヘッダ (図 7 参照) の暗号化フラグ (Encryption Flag) に基づいて判定される。暗号化対象でない場合は、復号処理は実行されず、処理は終了する。暗号化対象である場合は、暗号化フォーマットタイプをチェック (S 1 1 0 3) する。これはセキュリティヘッダ内の暗号化フォーマットタイプ (Encryption Format Type) の設定をチェックするものであり、図 8 で説明したコンテンツ全体を 1 つの暗号化態様としているか、各セクタに異なる鍵を用いた暗号化処理を行なっているかを判定する。

【0227】 暗号化フォーマットタイプ (Encryption Format Type) の設定値が 0 の場合は、コンテンツ全体を 1 つの暗号化態様としている場合である。この場合は、ステップ S 1 1 0 4 において、暗号化アルゴリズム (Encryption Algorithm) の判定を行なう。暗号化アルゴリズムは、シングル DES かトリプル DES (図 2 8 参照) かを設定しているものであり、シングル DES であると判定された場合は、1 つのコンテンツキー K_c (0) を適用して暗号化コンテンツの復号処理を実行 (S 1 1 0 6) する。トリプル DES であると判定された場合は、2 つのコンテンツキー K_c (0)、 K_c (1) を適用して暗号化コンテンツの復号処理を実行 (S 1 1 0 7) する。

【0228】 一方、ステップ S 1 1 0 3 で、暗号フォーマットタイプ (Encryption Format Type) の設定値が 1 の場合は、各セクタに異なる鍵を用いた暗号化処理を行

なっている場合である。この場合は、ステップ S 1 1 0 5 において、暗号化アルゴリズム (Encryption Algorithm) の判定を行なう。暗号化アルゴリズムは、シングル DES かトリプル DES (図 2 8 参照) かを設定しているものであり、シングル DES であると判定された場合は、各セクタ (s) に対応して設定されたコンテンツキー K_c (s) を各セクタに適用して暗号化コンテンツの復号処理を実行 (S 1 1 0 8) する。トリプル DES であると判定された場合は、2 つのコンテンツキー K_c (s)、 K_c ($s + 1 \bmod 32$) を適用して各セクタ毎の暗号化コンテンツの復号処理を実行 (S 1 1 0 9) する。

【0229】 セクタデータの復号処理の異なる処理態様を図 4 0 に示す。図 4 0 において、ステップ S 1 2 0 1 ~ S 1 2 0 8 は、図 3 9 の各ステップ S 1 1 0 1 ~ S 1 1 0 8 と同様である。ステップ S 1 2 0 9 ~ S 1 2 1 1 が図 3 9 とは異なる。

【0230】 ステップ S 1 2 0 5 において、暗号化アルゴリズムがトリプル DES であると判定された場合、ステップ S 1 2 0 9 においてセクタ No. (s) を判定し、 s が奇数である場合は、 $s = s - 1$ の更新を実行 (S 1 2 1 0) し、各セクタに適用する鍵を K_c (s)、 K_c ($s + 1$) としてトリプル DES による復号処理 (S 1 2 1 1) を実行する。

【0231】 以上、暗号化されて格納されたデータの復号処理を伴う再生処理は、図 3 5 ~ 図 4 0 を用いて説明したようなプロセスにより実行される。

【0232】 [データの暗号化書き込み処理] 次に、図 4 1 以下のフローを用いて、メディアに対するデータの暗号化書き込み処理プロセスの詳細を説明する。なお、データの暗号化態様は、上述したようにセクタ毎に異なる鍵で暗号化した態様と、コンテンツ全体を 1 つの暗号化鍵で暗号化した態様とがある。これらは、ヘッダ情報に設定される。図 4 1 のフローにおいて左側はデバイスの制御部、右側はデバイスのメモリインタフェースの処理である。

【0233】 まず制御部は、読み出し対象となる格納コンテンツに対応するヘッダ生成コマンドとヘッダ情報としてのパラメータをメモリインタフェースに送信する。(S 1 3 0 1)。

【0234】 メモリインタフェースはヘッダ生成コマンドを受信 (S 1 3 0 2) すると、ビジーフラグを 1 (ビジー) にセット (S 1 3 0 3) し、受信パラメータが許容値内であるかを判定 (S 1 3 0 4) する。メモリインタフェースは、予めヘッダに設定可能なパラメータ範囲を有しており、受信パラメータと比較し、受信パラメータが設定可能範囲を超えている場合は、ステップ S 1 3 1 0 においてヘッダ生成成功フラグを 0 (NG) に設定して処理を終了する。受信パラメータが許容値内である場合は、ヘッダの有効リポケーションリストバージョン

を0に設定 (S1305) し、リポケーションリストの非参照でのデータ処理を可能とする。有効リポケーションリストバージョンを0として設定するのは、自デバイスでの格納処理を行なったコンテンツについては正当なコンテンツであることが保証されているとの前提により、リポケーションリストの非参照でのデータ処理 (再生) を可能とする設定を行なうものである。

【0235】なお、書き込みコンテンツが例えば通信手段を介して外部から受信したコンテンツであり、受信コンテンツに識別子が付加され参照すべきリポケーションリストバージョンをヘッダに格納しておりデバイス内部のリポケーションリストとの照合が可能であれば、上記処理の代わりに、先に図35を用いて説明したファイル復号読み出し処理において実行されるステップS707～S709と同様のリポケーションリストを用いた識別子照合処理を行なってもよい。

【0236】次に、ステップS1306において、ヘッダ情報に基づいてコンテンツキーKc、コンテンツ改竄チェック値 (ICV) 生成鍵Kicv_contを生成、暗号化する。ステップS1306のコンテンツキーKc、コンテンツ改竄チェック値生成鍵Kicv_contを生成、暗号化処理の詳細を図43に示す。図43の処理は、デバイスのメモリインタフェースの暗号処理部320 (図4参照) において実行される。図43のフローについて説明する。

【0237】まず、暗号化コンテンツチェック値生成鍵Kicv_contを、例えば乱数に基づいて生成し、暗号化対象とし (S1401)、次に、ヘッダの暗号化フォーマットタイプ・フィールドの設定が0か否かを判定 (S1402) する。暗号化フォーマットが0である場合は、コンテンツ全体をセクタに係わらず1つの暗号化態様とする構成であり、暗号化フォーマットタイプ・フィールドの設定が1である場合は、前述の図27他で説明したセクタ単位の暗号化鍵を用いる方法である。セクタ単位の暗号化鍵を用いる場合は、ステップS1403に進み、セクタ毎に設定されたコンテンツキー (Kc (0)～Kc (31) (セクタ数32の場合)) を生成して暗号化対象とする。

【0238】ステップS1404で暗号化フォーマットが0であると判定された場合は、ステップS1404でさらに、ヘッダの暗号化アルゴリズムフィールドをチェックして1 (トリプルDES) か0 (シングルDES) であるかを判定する。シングルDESである場合は、ステップS1405で1つのコンテンツキー (Kc (0)) を生成して暗号化対象として加え、トリプルDESである場合は、ステップS1406で複数のコンテンツキー (Kc (0)、Kc (1)) を生成して暗号化対象として加える。

【0239】次に、ステップS1407において、ヘッダのコンテンツタイプフィールドの設定をチェックし、

設定が2または3 (メディア2の格納コンテンツ) でない場合は、ステップS1408で、メモリ部321 (図4参照) に格納された配送鍵Kdistでデータ、すなわち、コンテンツチェック値生成鍵Kicv_contと、1以上のコンテンツキーを暗号化する。

【0240】設定が2または3 (メディア2の格納コンテンツ) である場合は、ステップS1409でデータ、すなわち、コンテンツチェック値生成鍵Kicv_contと、1以上のコンテンツキーをメディア2の保存鍵Ksto (CBCモード) で暗号化する。この暗号化処理の詳細は、図32、図33、図34を用いて説明した通りである。

【0241】ステップS1409におけるメディア2の保存鍵によるコンテンツチェック値生成鍵Kicv_contと、1以上のコンテンツキーKcの暗号化処理について図44のフローを用いて説明する。図44のフローは、左側にデバイスのメモリインタフェース、右側にメディア2のコントローラ (図2参照) の処理を示している。

【0242】まず、デバイス側のメモリインタフェースは、暗号化対象データK (0)～K (n-1) (コンテンツチェック値生成鍵Kicv_contと、1以上のコンテンツキー) を設定 (S1501) し、メディア2との相互認証時に生成したセッションキーを適用し、メモリ部321に格納した初期値IV_keysを用いてDES-CBCモードによる暗号化対象データK (0)～K (n-1) の暗号化を実行し、データK' (0)～K' (n-1) を生成 (S1502) する。この暗号化処理は、先に説明した図32と同様の処理構成において実行される。次に、メモリインタフェースは、CBC暗号化初期化コマンドをメディア2コントローラに送信する。メディア2は、メディア2の内部に格納している初期値IV_keysをレジスタにセット (S1506) する。その後、メモリインタフェースは、各鍵を順次送信 (S1505) する。

【0243】メディア2コントローラは、データK' (i) を受信 (S1507) し、受信したデータK' (i) に対して、デバイスとの相互認証時に生成したセッションキーによってCBCモードでの復号処理を実行 (S1508) し、復号された鍵データ (ex. 複数のセクタ対応コンテンツキー) を取得 (S1509) する。次に、メディア2コントローラは、復号鍵データ列を、メディア2の保存鍵Kstoを用いたCBCモードによる暗号化処理を実行し、データ列K'' (i) を生成して、結果をデバイスに送信 (S1510) する。ステップS1507～S1510の処理は、先に説明した図34のDES-CBCモードによる処理に基づいて実行される。

【0244】デバイスのメモリインタフェースは、順次K'' (i) を受信し、すべてのデータを受信したことを

確認の後、CBC終了コマンドをメディア2コントローラに送信 (S1511~S1514) する。メディア2コントローラはCBC終了コマンドの受信によりレジスタをクリア (S1515) する。

【0245】デバイスのメモリインタフェースは、メディア2から受信したK" (0)~K" (n-1) をヘッダ格納用の暗号化鍵データとする。上記処理により、デバイスは、ヘッダに格納する暗号化されたコンテンツキーKc、コンテンツチェック値生成鍵Kicv_contを取得することができる。

【0246】図41に戻り、ファイルの暗号化書き込み処理の説明を続ける。ステップS1306において、上述のヘッダ格納鍵の生成、暗号化が終了すると、メモリインタフェースは生成したヘッダデータに基づく改竄チェック値ICVを生成 (S1307) する。セキュリティヘッダのチェック値であるICV_shは、メモリ部321 (図4参照) に格納された初期値IVshと、セキュリティヘッダ改竄チェック値生成鍵Kicv_shを用いて、先に図14を用いて説明したICV生成構成に基づいて生成される。次に、ステップS1308で、生成されたヘッダを書き込みヘッダとして内部に保存し、ステップS1309でヘッダ生成成功フラグを1 (成功) としてビジーフラグを0 (待機) として処理を終了する。

【0247】一方、制御部側は、ステップS1312でステータス読み出しコマンドをメモリインタフェースに送信し、ビジーフラグが0 (待機) (S1313) であり、ヘッダ生成成功フラグが1 (成功) (S1314) となったことを条件として、バッファからヘッダを読み出し、通常のファイルとしてメディアに保存 (S1315) 後、次の処理 (図42) に進む。

【0248】図42のステップS1321において、制御部は、書き込み対象のコンテンツファイルをセクタに分割する。分割されたデータをD (1)~D (k) とする。制御部は、次に各データD (i) の書き込みセクタS (i) を設定して、メモリインタフェースにセクタS (i) の暗号化書き込みコマンドと、データD (i) を順次送信 (S1321~S1324) する。メモリインタフェースは、セクタS (i) 暗号化書き込みコマンドを受信 (S1325) すると、ビジーフラグを1 (ビジー) に設定 (S1326) し、ヘッダ生成成功フラグが1 (成功) である (S1327) ことを条件として次ステップに進む。

【0249】次に、メモリインタフェースは、受信セクタS (i) が内部メモリか、外部メモリであるかを判定 (S1328) し、外部メモリである場合は、メディア1かメディア2のセットフラグが1 (メディアが有効にセットされていることを示す) であるかを判定 (S1329) し、セットフラグが1である場合には、さらにブロックパーミッション・テーブル (BPT) を参照し

て、BPTが書き込み対象であるセクタS (i) を書き込み許可対象ブロックとして設定しているかを判定 (S1330) する。BPTに書き込み許可ブロックの設定がある場合には、セクタに対応して設定する誤り訂正符号を生成 (S1331) する。

【0250】次に、その書き込みセクタがICV設定セクタであるか否かをヘッダ情報 (ICVフラグ) に基づいて判定 (S1332) し、ICV対象である場合は、コンテンツICV生成鍵Kicv_contに基づいてセクタデータに対するICVを生成 (S1333) する。

【0251】次に、メモリインタフェースは、ヘッダ情報に基づくデータの暗号化を実行 (S1334) する。ステップS1334のデータ部暗号化処理の詳細を図45を用いて説明する。この暗号化処理はデバイスのメモリインタフェースの暗号処理部320 (図4参照) において実行される。

【0252】まず、暗号化対象のデータ格納セクタ位置をs ($0 \leq s \leq 31$ (セクタ数32の場合)) とする (S1601)。次にそのセクタが暗号化対象であるかをチェック (S1602) する。このチェックは、セキュリティヘッダ (図7参照) の暗号化フラグ (Encryption Flag) に基づいて判定される。暗号化対象でない場合は、暗号化処理は実行されず、処理は終了する。暗号化対象である場合は、暗号化フォーマットタイプをチェック (S1603) する。これはセキュリティヘッダ内の暗号化フォーマットタイプ (Encryption Format Type) の設定をチェックするものであり、図8で説明したコンテンツ全体を1つの暗号化態様としているか、各セクタに異なる鍵を用いた暗号化処理を行なっているかを判定する。

【0253】暗号化フォーマットタイプ (Encryption Format Type) の設定値が0の場合は、コンテンツ全体を1つの暗号化態様としている場合である。この場合は、ステップS1604において、暗号化アルゴリズム (Encryption Algorithm) の判定を行なう。暗号化アルゴリズムは、シングルDESかトリプルDES (図28参照) かを設定しているものであり、シングルDESであると判定された場合は、1つのコンテンツキーKc (0) を適用して暗号化コンテンツの暗号化処理を実行 (S1606) する。トリプルDESであると判定された場合は、2つのコンテンツキーKc (0)、Kc (1) を適用して暗号化コンテンツの暗号化処理を実行 (S1607) する。

【0254】一方、ステップS1603で、暗号フォーマットタイプ (Encryption Format Type) の設定値が1の場合は、各セクタに異なる鍵を用いた暗号化処理を行なう場合である。この場合は、ステップS1605において、暗号化アルゴリズム (Encryption Algorithm) の判定を行なう。暗号化アルゴリズムは、シングルDESかトリプルDES (図28参照) かを設定しているもの

であり、シングルDESであると判定された場合は、各セクタ (s) に対応して設定されたコンテンツキー $K_c(s)$ を各セクタに適用して暗号化コンテンツの暗号化処理を実行 (S1608) する。トリプルDESであると判定された場合は、2つのコンテンツキー $K_c(s)$ 、 $K_c(s+1 \bmod 32)$ を適用して各セクタ毎の暗号化処理を実行 (S1609) する。

【0255】セクタデータの復号処理の異なる処理態様を図46に示す。図46において、ステップS1701～S1708は、図45の各ステップS1601～S1608と同様である。ステップS1709～S1711が図45とは異なる。

【0256】ステップS1705において、暗号化アルゴリズムがトリプルDESであると判定された場合、ステップS1709においてセクタNo. (s) を判定し、sが奇数である場合は、 $s=s-1$ の更新を実行 (S1710) し、各セクタに適用する鍵を $K_c(s)$ 、 $K_c(s+1)$ としてトリプルDESによる復号処理 (S1711) を実行する。

【0257】図42に戻り、ファイルの暗号化書き込み処理フローの説明を続ける。上述の処理によってデータ部の暗号化処理ステップ (S1334) が終了すると、データ部に対する誤り訂正符号を生成 (S1335) し、暗号化されたデータD(i)とセクタデータに対応する改竄チェック値ICVと、誤り訂正符号を持つ冗長部をメディアに書き込み (S1336)、書き込み成功フラグを1 (成功) にセット (S1337) し、ビジーフラグを0 (待機) に設定 (S1339) する。

【0258】なお、書き込み対象データがBPTによる管理のなされていない内部メモリ内への書き込み処理である場合は、ステップS1329、S1330はスキップする。ステップS1329、S1330の判定がNoである場合、すなわちメディアのセットフラグが1でない場合、または、BPTにセクタS(i)の書き込み許可が設定されていない場合には、ステップS1338に進み、書き込みエラーとして書き込み成功フラグを0にセットする。

【0259】また、制御部は、ステップS1341～S1345において、メモリインタフェースのステータスを読み出して、ビジーフラグが0の状態において、書き込み成功フラグが1であることを条件としてアドレスを順次インクリメントして、書き込みデータを順次メモリインタフェースに送信する。すべての処理が終了すると、ファイル割り当てテーブルの更新処理を実行 (S1346) し、更新したファイル割り当てテーブルを更新コマンドとともにメモリインタフェースに送信 (S1347) し、メモリインタフェースはコマンドに従ってファイル割り当てテーブルの書き込み処理を実行 (S1340) する。

【0260】以上の、図41～図46によって説明した

処理により、データの暗号化、メディアに対する格納処理が実行される。

【0261】[リボケーションリストの更新] 次に、不正なメディアやコンテンツの失効情報としてのリボケーションリストの更新処理について説明する。前述したように、本発明におけるリボケーションリストは、複数の種類 (ex. メディア、コンテンツ) の識別子 (ID) から構成される。コンテンツやメディアの失効情報であるリボケーションリスト (Revocation List) に複数の種類のIDを設け、それぞれの照合を異なる動作として行うことによって、1つのリボケーションリストで複数の種類のコンテンツ、メディアを排除することが可能となる。メディアの挿入時やコンテンツの読み出し時にメモリ・インターフェース部において、利用メディアまたは利用コンテンツの識別子 (ID) と、リボケーションリストのリスティングIDとの照合を実行することにより、不正なメディアの使用や不正なコンテンツの読み出しを禁止することができる。

【0262】先に説明したように、リボケーションリストには、リボケーションリストバージョン (Revocation List Version) が設定され、新たな不正なメディアやコンテンツの失効情報を追加した場合等にリボケーションリストは更新される。

【0263】リボケーションリストの更新処理フローを図47に示す。図47において、左側はデバイスの制御部、右側はデバイスのメモリインタフェースである。

【0264】まず、制御部は更新用のリボケーションリストを通信部201 (図2参照) から受信する (S1801) と、更新用リボケーションリストチェックコマンドと、受信した更新用リボケーションリストをメモリインタフェースに送信 (S1802) する。

【0265】メモリインタフェースは、更新用リボケーションリストチェックコマンドと、更新用リボケーションリストを制御部から受信 (S1803) すると、ビジーフラグを1 (ビジー) に設定 (S1804) し、リボケーションリストの改竄チェック値 (ICV) 生成鍵 K_{icv_rl} を生成 (S1805) する。

【0266】リボケーションリストの改竄チェック用の改竄チェック値 (ICV) 生成鍵 K_{icv_rl} は、予めデバイス内に格納されたりボケーションリスト (Revocation List) のICV鍵を生成するマスター鍵: MK_{icv_rl} と、リボケーションリスト (Revocation List) のICV鍵を生成する時の初期値: IV_{icv_rl} と、リボケーションリストの属性情報に含まれるリボケーションリスト・バージョン (Version) に基づいて生成する。具体的には、改竄チェック値 (ICV) 生成鍵 $K_{icv_rl} = DES(E, MK_{icv_rl}, Version, IV_{icv_rl})$ に基づいて改竄チェック値 (ICV) 生成鍵が生成される。前記式の意味は、バージョン (Version) と初期値 (IV_{icv_rl}) の排他論理和にマスター鍵: MK_{icv_rl}

v_{rl}によるDESモードでの暗号化処理を実行するという意味である。

【0267】次にメモリインタフェースは生成した改竄チェック値(ICV)生成鍵K_{icv_{rl}}を用いてリポケーションリストのICV'を生成(S1806)し、予めリポケーションリスト内に格納された正しいICV値と照合ICV' = ICV?を実行(S1807)する。なお、ICV'の生成処理は、前述の図14で説明したDESモードに基づいて、初期値IV_{rl}を用い、生成した改竄チェック値(ICV)生成鍵K_{icv_{rl}}を適用した処理によって行われる。

【0268】ICV' = ICVである場合(S1807でYes)は、更新用リポケーションリストが改竄のない正当なものであると判定され、ステップS1808に進み、現在セットされているリポケーションリストのバージョン(i)と更新用リポケーションリストのバージョン(j)を比較(S1809)し、更新用リポケーションリストのバージョンが新しい場合には、更新用リポケーションリストの有効フラグを1に設定(S1810)し、ビジーフラグを0にセット(S1811)して処理を終了する。

【0269】一方、制御部側は、ステータス読み出しコマンドをメモリインタフェースに送信(S1812)し、ビジーフラグが0となった(S1813)ことを確認し、更新用リポケーションリスト有効フラグが1(S1814)である場合に、更新用リポケーションリストを通常のファイルとして内部メモリに保存(S1815)する。コンテンツの処理、メディアの装着時のチェックの際には、内部メモリに格納されたりポケーションリストが読み出される。

【0270】以上、特定の実施例を参照しながら、本発明について詳解してきた。しかしながら、本発明の要旨を逸脱しない範囲で当業者が該実施例の修正や代用を成し得ることは自明である。すなわち、例示という形態で本発明を開示してきたのであり、限定的に解釈されるべきではない。本発明の要旨を判断するためには、冒頭に記載した特許請求の範囲の欄を参酌すべきである。

【0271】

【発明の効果】以上、説明したように、本発明のデータ再生装置、データ記録装置、およびデータ再生方法、データ記録方法、並びにプログラム提供媒体によれば、リポケーションリストに異なるカテゴリのメディア、コンテンツ双方の識別子を格納して、不正メディア、不正コンテンツ双方の排除を1つのリポケーションリストに基づいて実行可能な構成としたので、デバイス側で保持するリポケーションリストが減少し、メディアの装着時、コンテンツ利用時のデバイス側での処理が軽減される。

【0272】さらに、本発明のデータ再生装置、データ記録装置、およびデータ再生方法、データ記録方法、並びにプログラム提供媒体によれば、コンテンツやメディア

の失効情報であるリポケーションリストに複数種類のリポーク対象ID、すなわちメディアID、コンテンツIDを設け、それぞれのデバイス側での照合処理を異なる動作、例えばメディア装着時にはメディア識別子との照合、コンテンツ再生時にはコンテンツ識別子との照合を行うことによって1つのリポケーションリストで複数のコンテンツ、メディアを排除することが可能となり、メディアの挿入時やコンテンツの読み出し時のメモリ・インターフェース部におけるリポケーションリストの参照処理が効率的に実行され、不正なメディアの使用や不正なコンテンツの読み出しを効率的に禁止することができる。

【図面の簡単な説明】

【図1】本発明のデータ処理装置の使用概念を説明する図である。

【図2】本発明のデータ処理装置のデバイスおよびメディアの構成を示す図である。

【図3】本発明のデータ処理装置のメモリ格納データ構成を示す図である。

【図4】本発明のデータ処理装置にデバイスのメモリインタフェースの詳細構成を示す図である。

【図5】本発明のデータ処理装置におけるメモリインタフェースのステータスレジスタのデータ構成を示す図である。

【図6】本発明のデータ処理装置におけるメディアに格納されるデータの詳細構成を示す図である。

【図7】本発明のデータ処理装置においてメディアに格納されるコンテンツに対応して設定されるセキュリティヘッダの構成を説明する図である。

【図8】本発明のデータ処理装置におけるデータ暗号化の2つの態様を説明する図である。

【図9】本発明のデータ処理装置におけるリポケーションリストの構成を示す図である。

【図10】本発明のデータ処理装置におけるブロック・パーミッション・テーブル(BPT)について説明する図である。

【図11】本発明のデータ処理装置におけるメディア1製造時のBPT格納処理フローを示す図である。

【図12】本発明のデータ処理装置におけるメディア2製造時のBPT格納処理フローを示す図である。

【図13】本発明のデータ処理装置におけるブロック・パーミッション・テーブル(BPT)の具体例について説明する図である。

【図14】本発明のデータ処理装置における改竄チェック値生成処理構成について説明する図である。

【図15】本発明のデータ処理装置における改竄チェック値検証処理フローについて説明する図である。

【図16】本発明のデータ処理装置におけるデバイス起動時フローを示す図である。

【図17】本発明のデータ処理装置におけるファイル割

り当てテーブルの構成例について説明する図である。

【図 18】本発明のデータ処理装置におけるメディア 1 認識時フロー（その 1）を示す図である。

【図 19】本発明のデータ処理装置におけるメディア 1 認識時フロー（その 2）を示す図である。

【図 20】本発明のデータ処理装置におけるメディア 2 認識時フロー（その 1）を示す図である。

【図 21】本発明のデータ処理装置におけるメディア 2 認識時フロー（その 2）を示す図である。

【図 22】本発明のデータ処理装置においてデバイス・メディア間において実行される相互認証処理シーケンスを示す図である。

【図 23】本発明のデータ処理装置における相互認証・鍵共有処理フロー（その 1）を示す図である。

【図 24】本発明のデータ処理装置における相互認証・鍵共有処理フロー（その 2）を示す図である。

【図 25】本発明のデータ処理装置におけるファイルの読み出し処理フローを示す図である。

【図 26】本発明のデータ処理装置におけるファイルの書き込み処理フローを示す図である。

【図 27】本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様を説明する図である。

【図 28】本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様として適用可能なトリプル DES を説明する図である。

【図 29】本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様を説明する図である。

【図 30】本発明のデータ処理装置におけるメモリに格納されたデータの暗号化処理態様を説明する図である。

【図 31】本発明のデータ処理装置におけるセクタ対応改竄チェック値の格納処理態様を説明する図である。

【図 32】本発明のデータ処理装置におけるセクタ対応コンテンツキー他の鍵の暗号化処理例を説明する図である。

【図 33】本発明のデータ処理装置におけるセクタ対応コンテンツキー他の鍵の復号処理例を説明する図である。

【図 34】本発明のデータ処理装置におけるセクタ対応コンテンツキー他の鍵のデバイス・メディア間における処理例を説明する図である。

【図 35】本発明のデータ処理装置におけるファイルの復号読み出し処理フロー（その 1）を示す図である。

【図 36】本発明のデータ処理装置におけるファイルの復号読み出し処理フロー（その 2）を示す図である。

【図 37】本発明のデータ処理装置におけるコンテンツキー他の復号処理フローを示す図である。

【図 38】本発明のデータ処理装置におけるコンテンツキー他のメディアの保存鍵による復号処理フローを示す図である。

【図 39】本発明のデータ処理装置におけるセクタデー

タの復号処理フロー（その 1）を示す図である。

【図 40】本発明のデータ処理装置におけるセクタデータの復号処理フロー（その 2）を示す図である。

【図 41】本発明のデータ処理装置におけるファイルの暗号化書き込み処理フロー（その 1）を示す図である。

【図 42】本発明のデータ処理装置におけるファイルの暗号化書き込み処理フロー（その 2）を示す図である。

【図 43】本発明のデータ処理装置におけるコンテンツキー他の暗号化処理フローを示す図である。

【図 44】本発明のデータ処理装置におけるコンテンツキー他のメディアの保存鍵による暗号化処理フローを示す図である。

【図 45】本発明のデータ処理装置におけるセクタデータの暗号化処理フロー（その 1）を示す図である。

【図 46】本発明のデータ処理装置におけるセクタデータの暗号化処理フロー（その 2）を示す図である。

【図 47】本発明のデータ処理装置におけるリポケーションリストの更新処理フローを示す図である。

【符号の説明】

101	システム運営者
102	デバイス
103	メディア
200	デバイス
201	通信部
202	入力部
203	表示部
204	デバイスコントローラ
205	制御部
207	メモリ部
300	メモリインタフェース（I/F）部
210	メディア 1
211	制御部
212	メモリ部
230	メディア 2
231	コントローラ
232	メモリ部
233	制御部
234	メモリインタフェース（I/F）部
235	内部メモリ
236	暗号処理部
301	ステータスレジスタ
302	コマンドレジスタ
303	アドレスレジスタ
304	カウンタレジスタ
305	コントロールレジスタ
306	送受信制御部
307	送信バッファメモリ
308	受信バッファメモリ
309	送信レジスタ
310	受信レジスタ

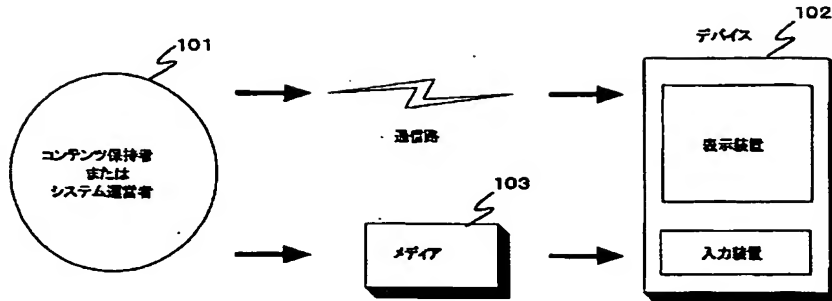
61

62

320 暗号処理部
321 メモリ部
323 ECC回路

324 外部メモリ入出力インタフェース
325 内部メモリ入出力インタフェース

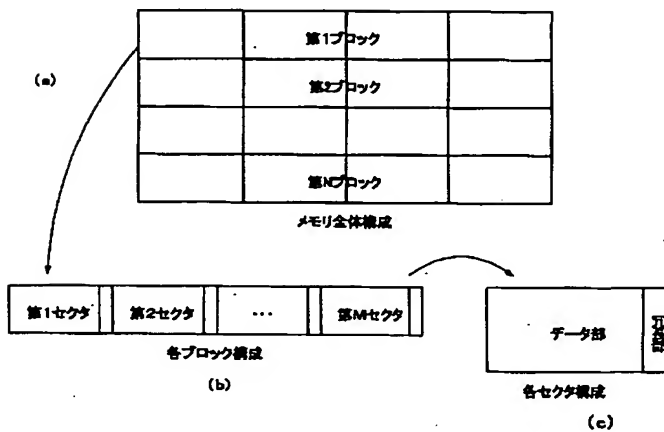
【図1】



【図9】

Revocation List ID
Revocation List Version
Number of Media1 ID
Media1 ID(0)
.....
Media1 ID(L-1)
Number of Media2 ID
Media2 ID(0)
.....
Media2 ID(M-1)
Number of Contents ID
Contents ID(0)
.....
Contents ID(N-1)
ICV of Revocation List

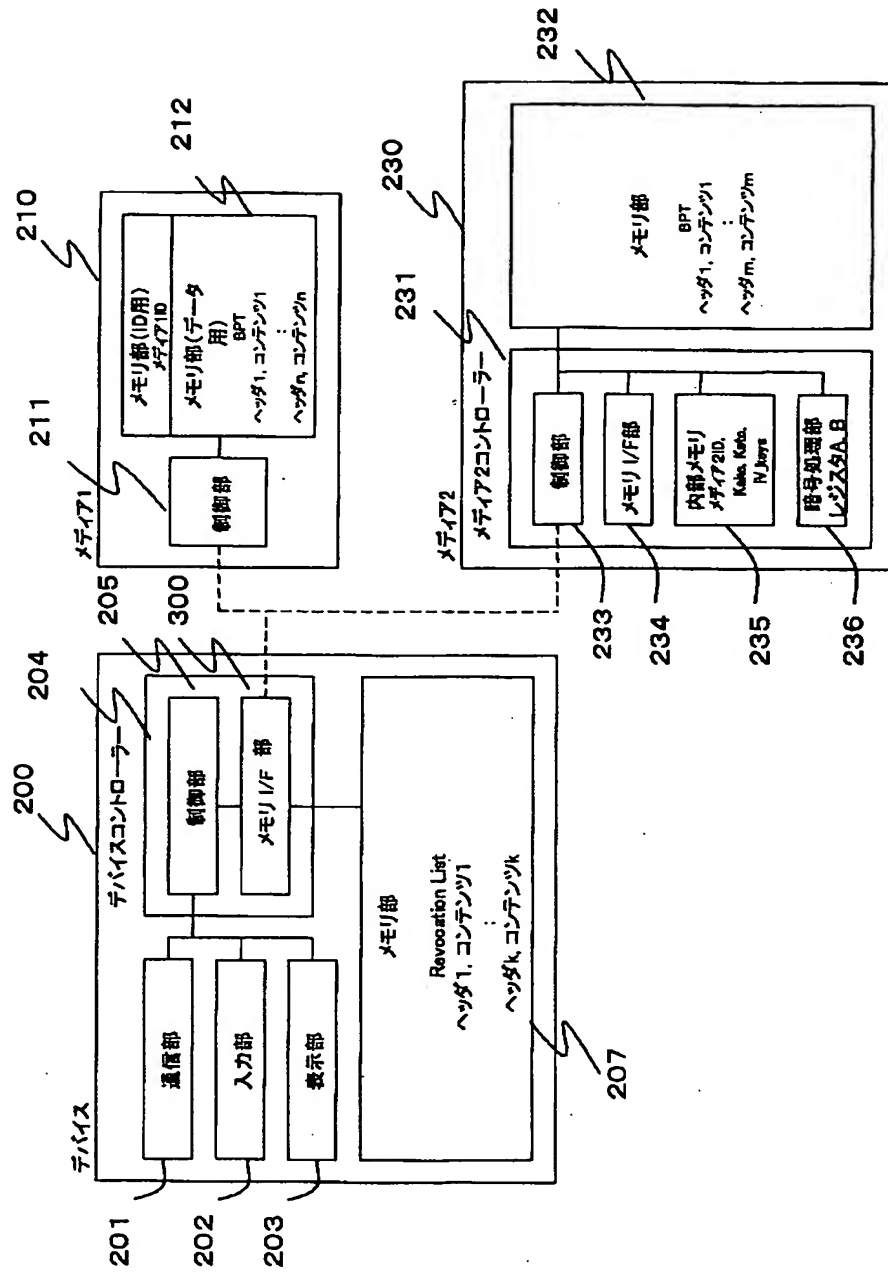
【図3】



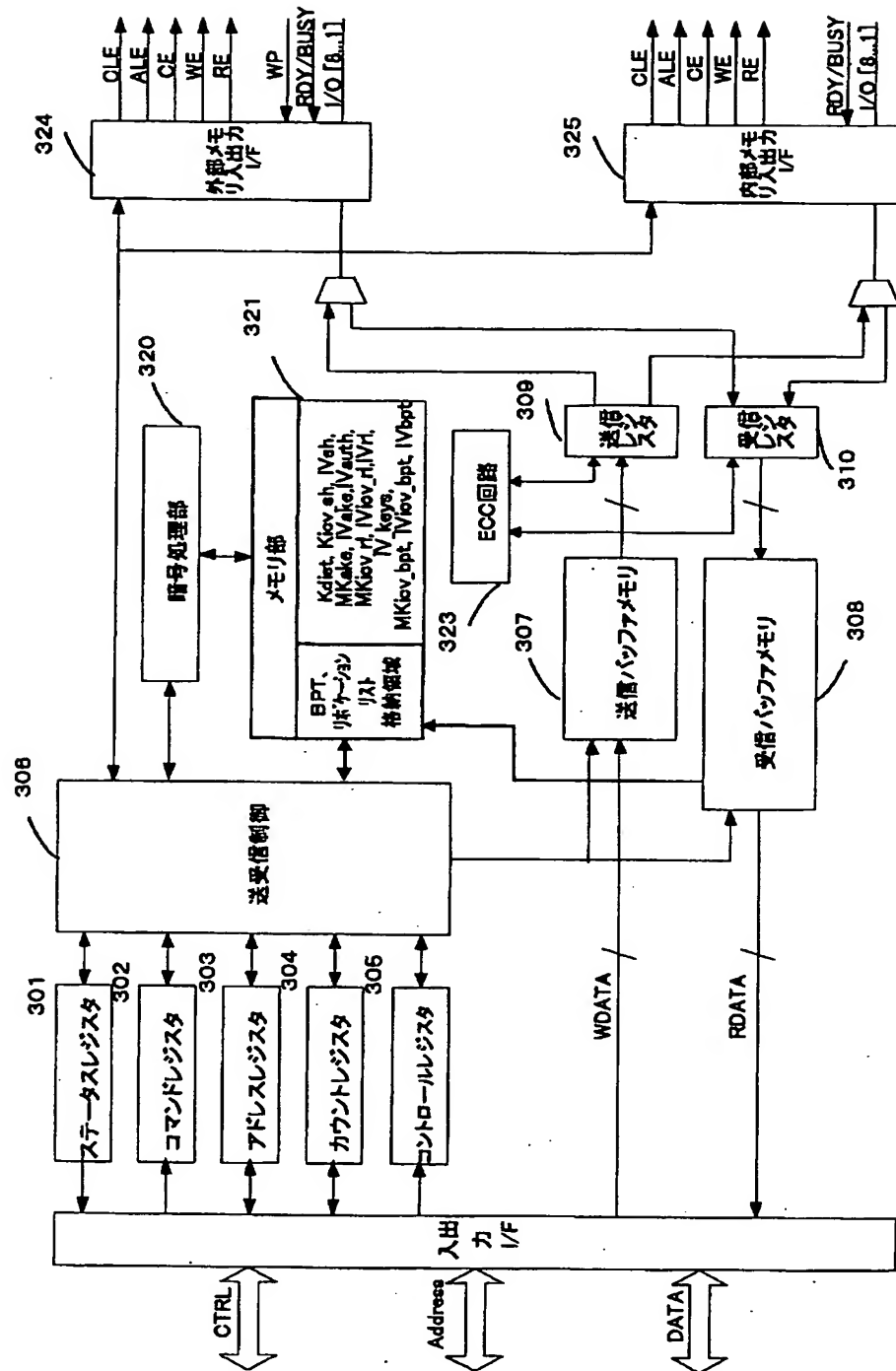
【図17】

ディレクトリ	ファイル名	格納セクタ
/	Ah	1 ~ 10
/	Acont	21 ~ 100
/dir_a	Bh	101 ~ 110
/dir_a	Bcont	111 ~ 350
/dir_a/dir_x	C	401 ~ 450
/dir_b	D	501 ~ 580
/dir_c	Eh	601 ~ 810
:	:	:
/dir_o	Zcont	5001 ~ 5340

【図2】



【図4】

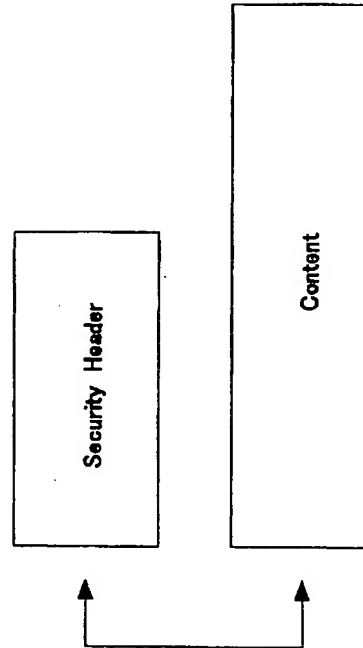


【図5】

bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
置換置換 List有効フラグ	置換置換 List有効フラグ	ヘッダ生成 成功フラグ	ヘッダセッ ト成功フラグ	メディア2 有効フラグ	メディア1 有効フラグ	メディア2 セットフラ グ	メディア1 セットフラ グ	書き込み成 功フラグ	読み出し成 功フラグ	ビジーフラ グ

【図6】

Block #1	Security Header 1
Block #2	Content 1
Block #3	Content 1
Block #4	Security Header 2
Block #5	Content 2
Block #6	Security Header 3
Block #7	Content 3
...	...



(a)

(b)

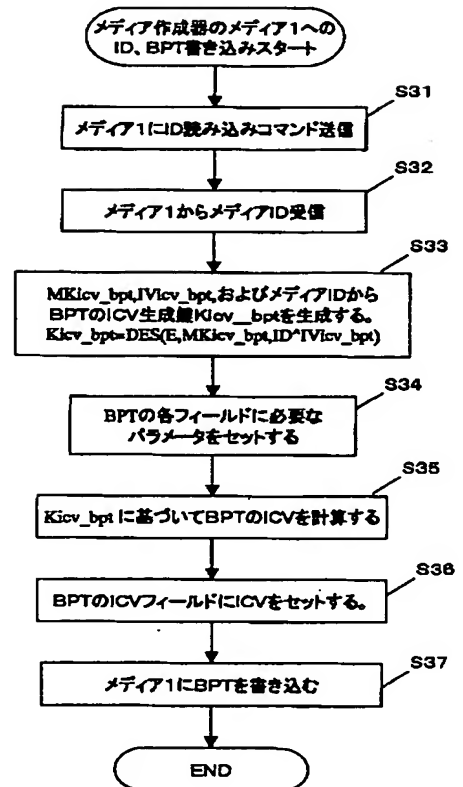
【図 7】

Format Version
Content ID
Content Type
Data Type
Encryption Algorithm
Encryption Mode
Encryption Format Type
Encryption Flag
ICV Flag
Ko_Encrypted 0
...
Ko_Encrypted 31
Kicv_cont_encrypted
Valid Revocation List version
ICV of Security Header

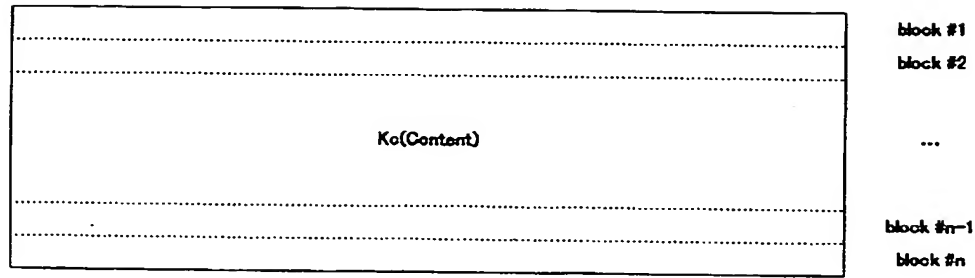
【図 10】

Format Version
BPT ID
Number of Blocks
Block #1 Permission Flag
...
Block #n Permission Flag
ICV of BPT

【図 11】



【図8】

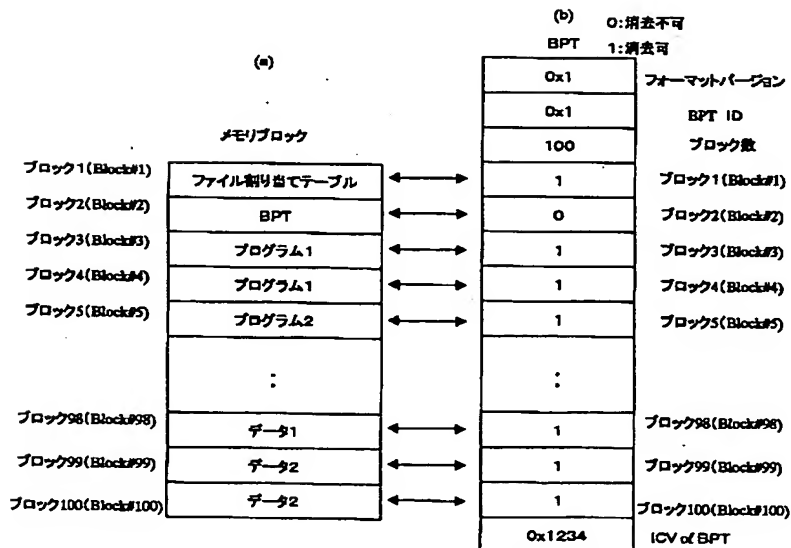


sector # 1	...	sector #m	
Ksao_1(Content_sao(1))	...	Ksao_m(Content_sao(m))	block #1
Ksao_1(Content_sao(m+1))	...	Ksao_m(Content_sao(2m))	block #2
...			...
Ksao_1(Content_sao((n-2)m+1))	...	Ksao_m(Content_sao((n-1)m))	block #n-1
Ksao_1(Content_sao((n-1)m+1))	...	Ksao_m(Content_sao(nm))	block #n

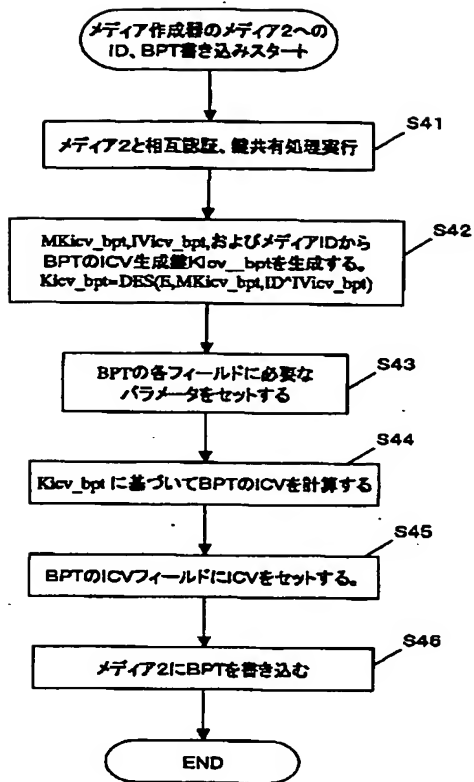
↑
各ブロックのセクタ#1は
Ksao_1 で暗号化

↑
各ブロックのセクタ#mは
Ksao_m で暗号化

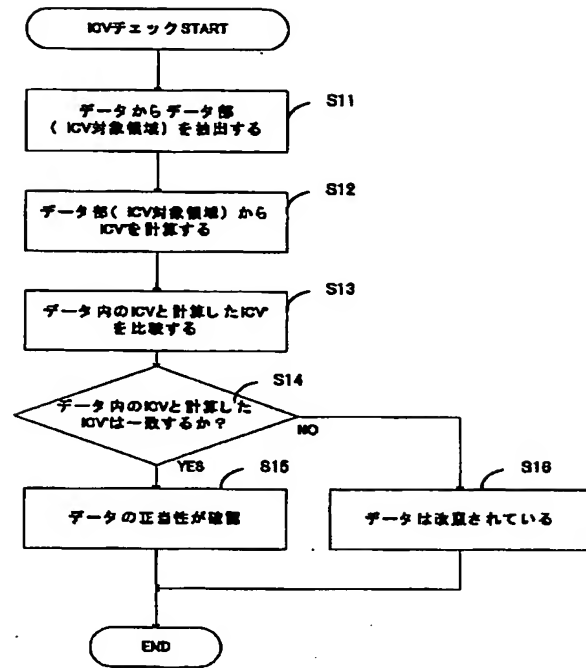
【図13】



【図12】

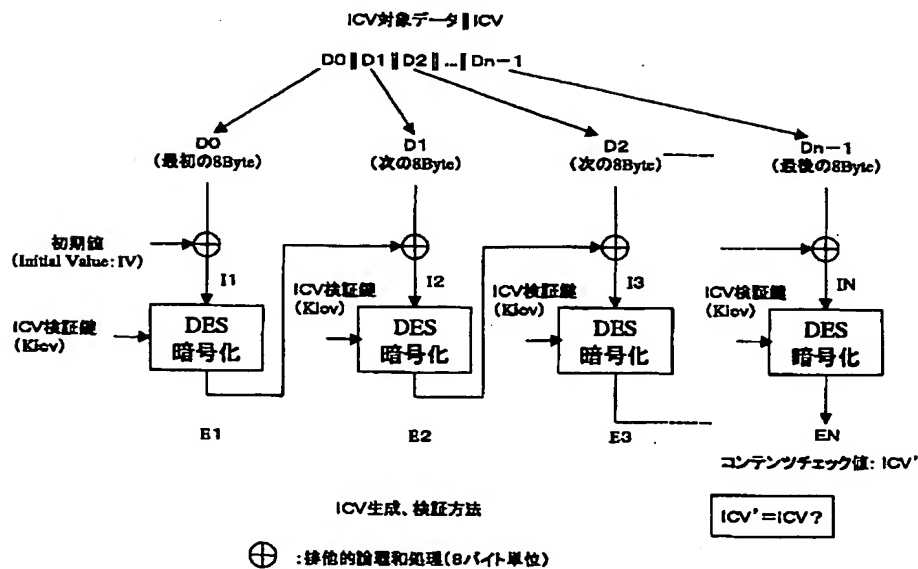


【図15】

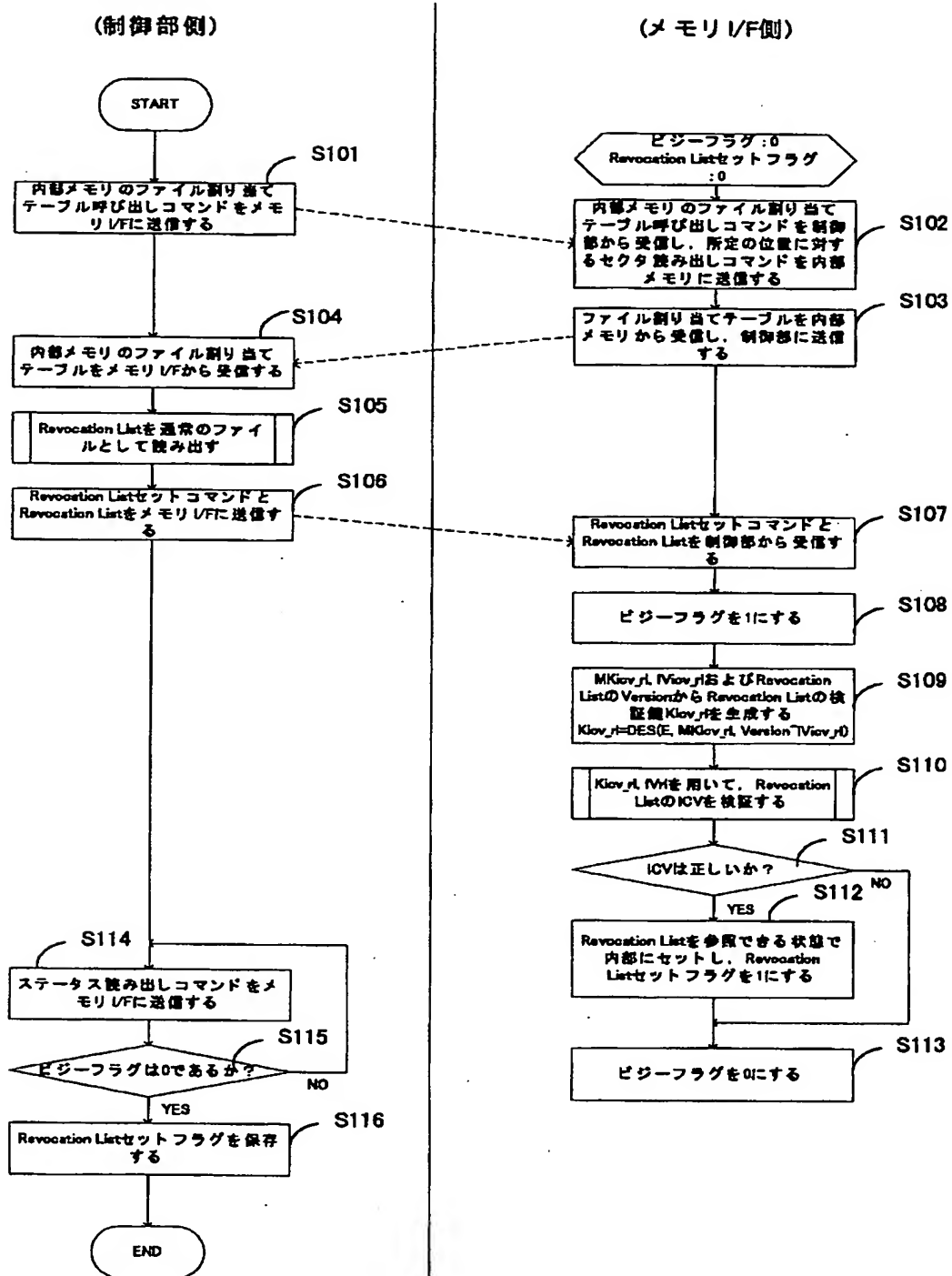


ICVチェック

【図14】

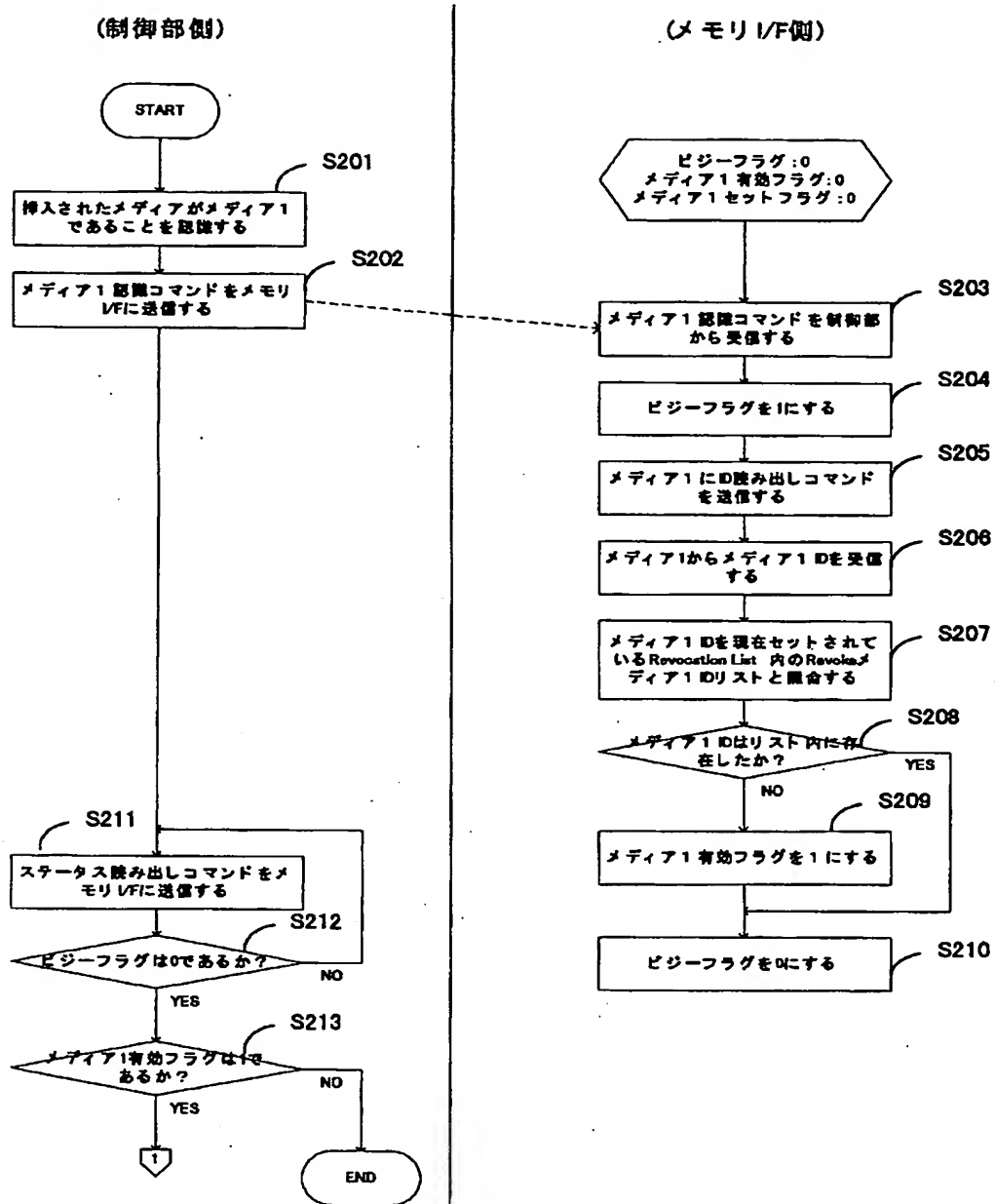


【図 16】



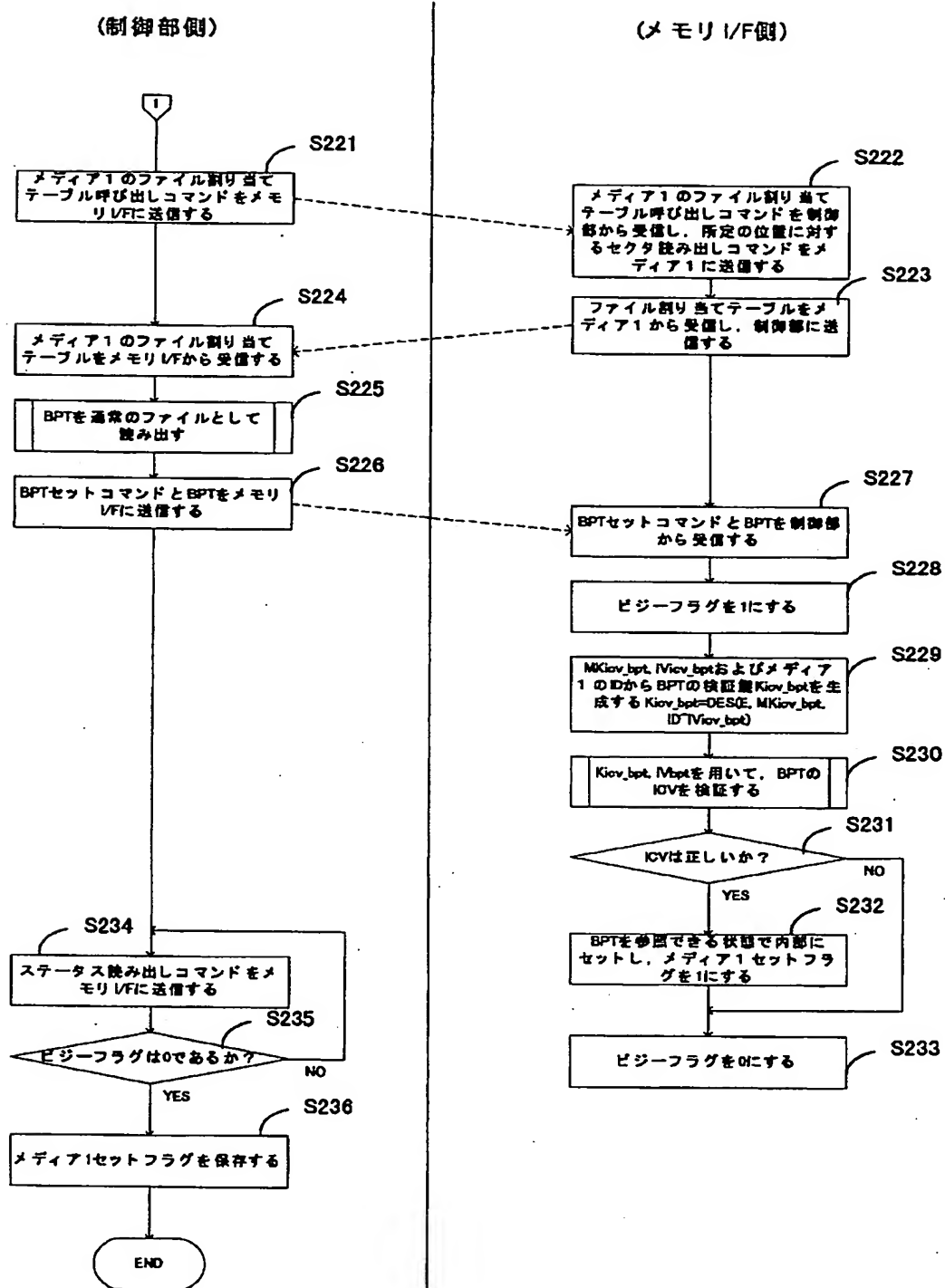
デバイス起動時フロー

【図 18】

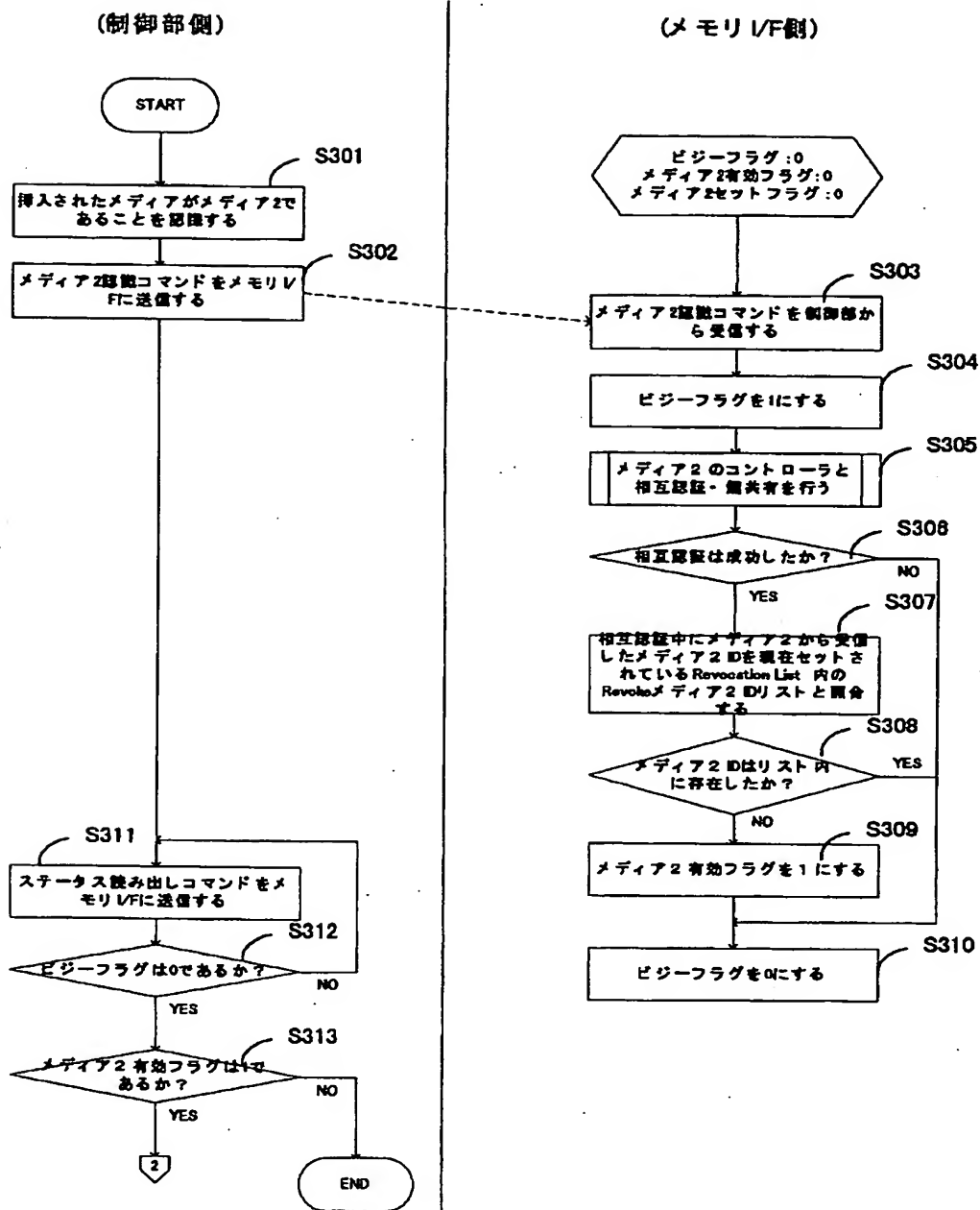


メディア1 認識時フロー

【図 19】

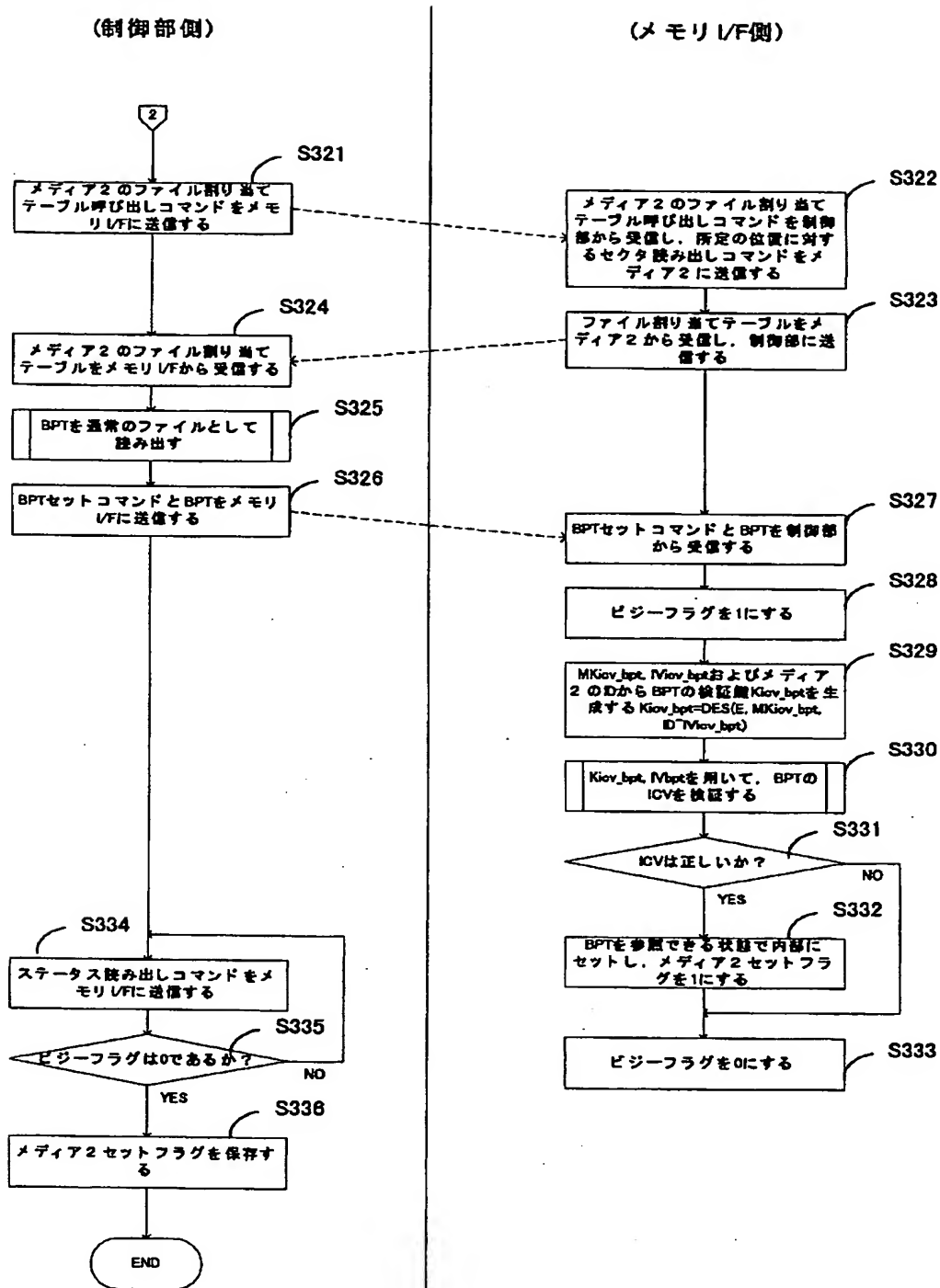


【図20】



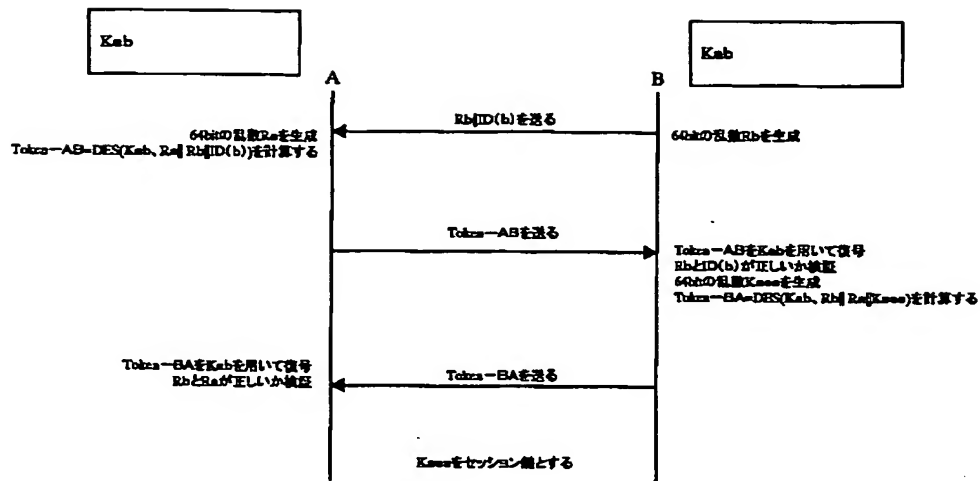
メディア2 認識時フロー

【図 21】



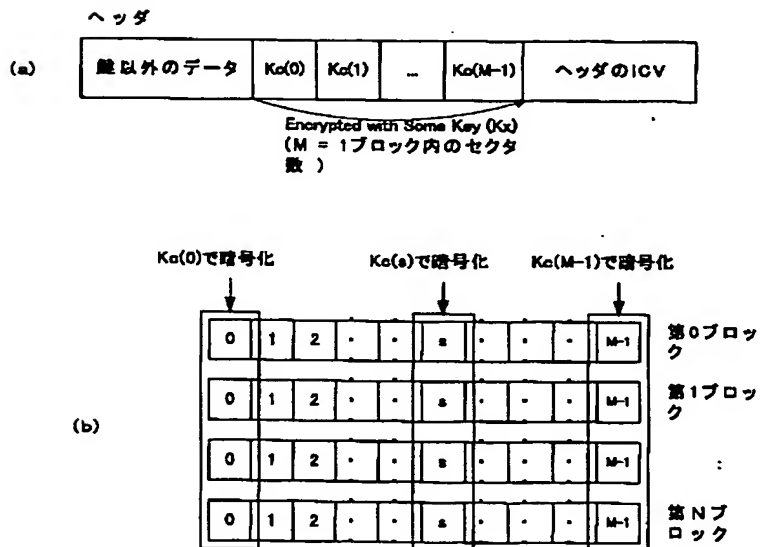
メディア2 認識時フロー (cont.)

【図22】

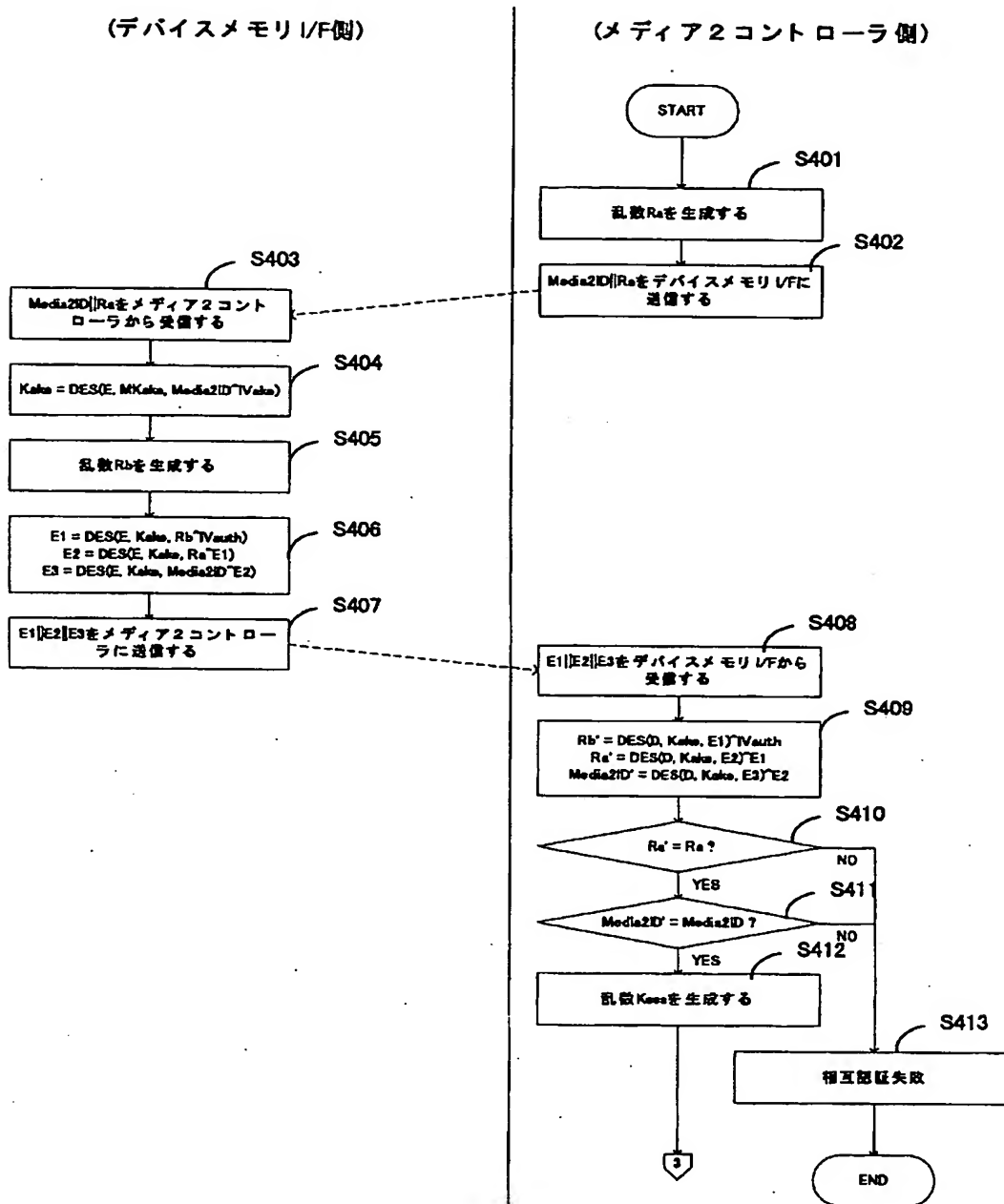


ISO/IEC 9798-2 対称暗号技術を用いた相互認証および鍵共有方式

【図27】

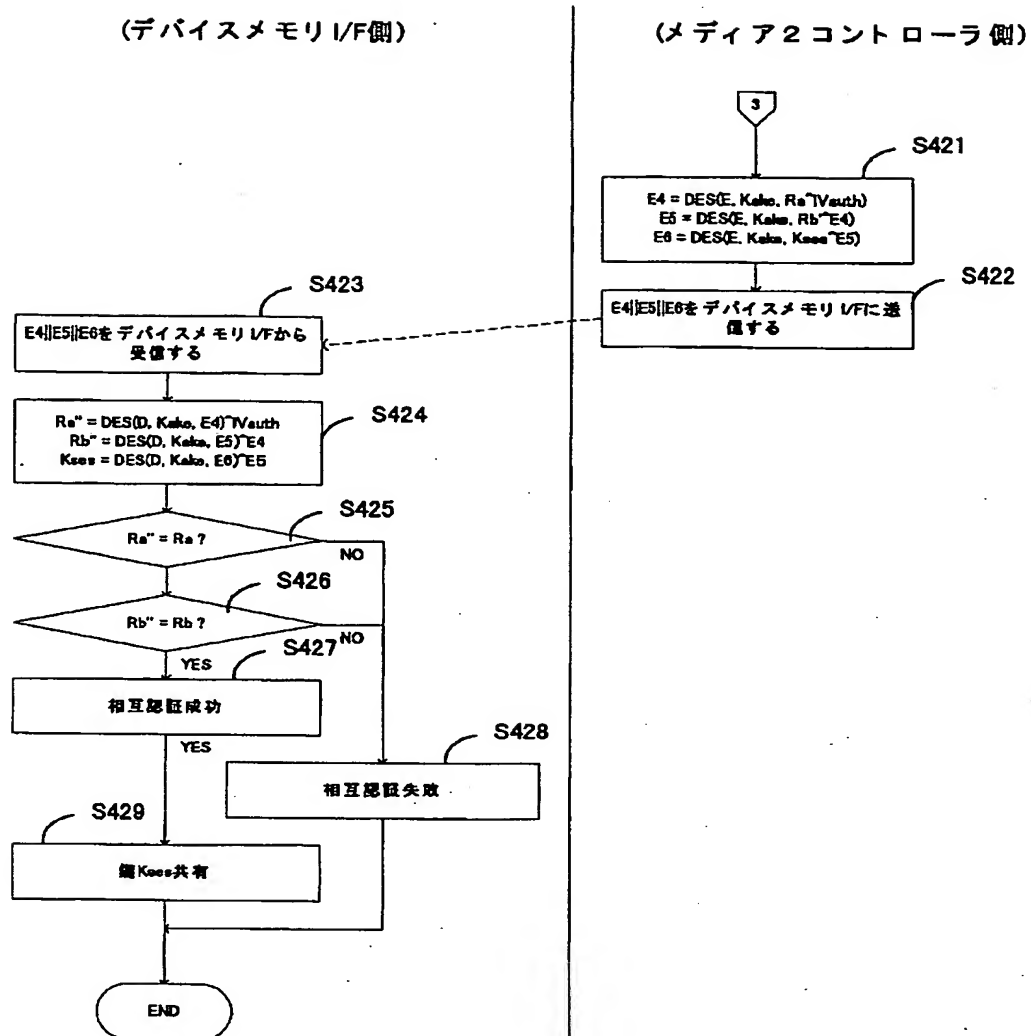


【図 23】



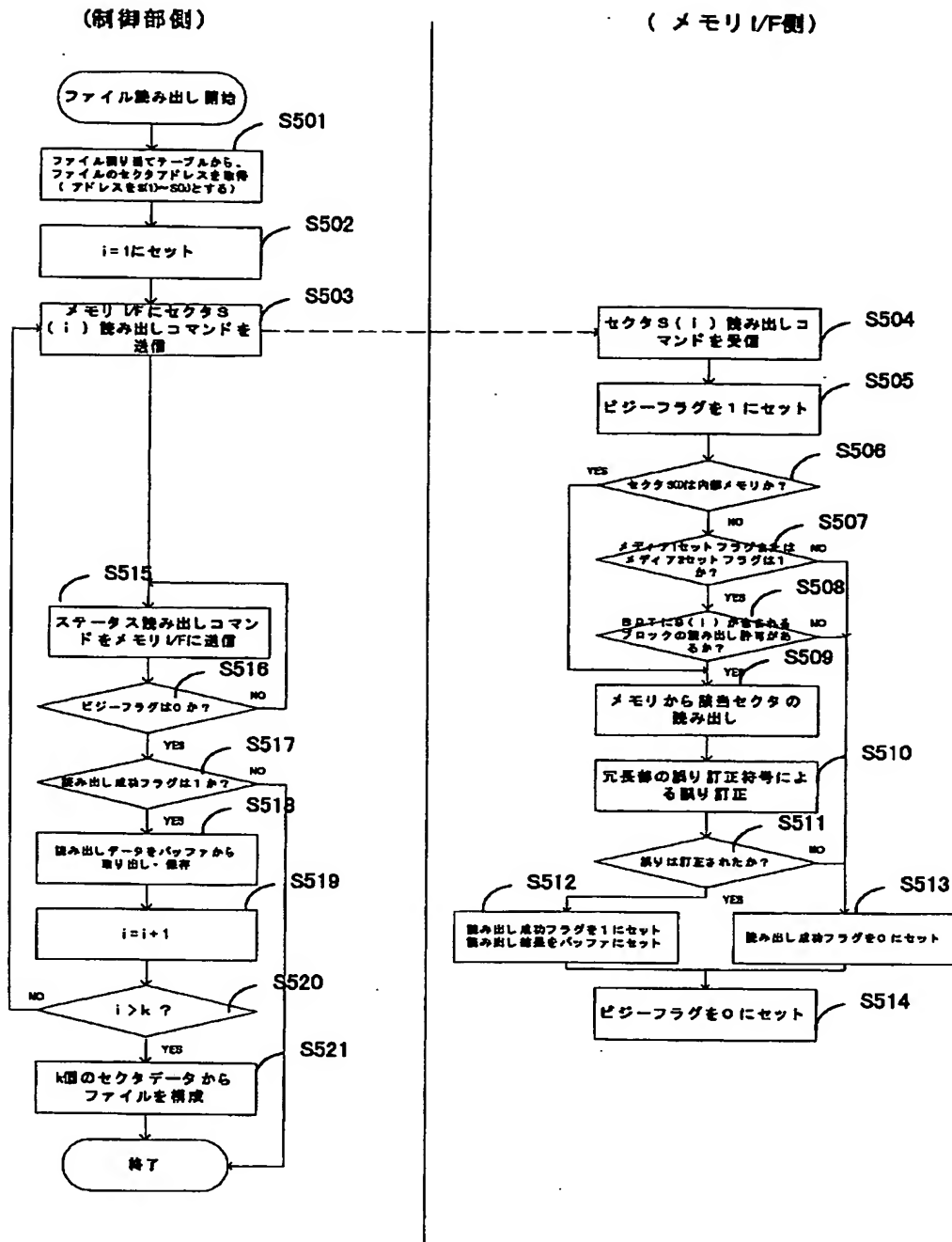
相互認証・鍵共有フロー

【図 24】



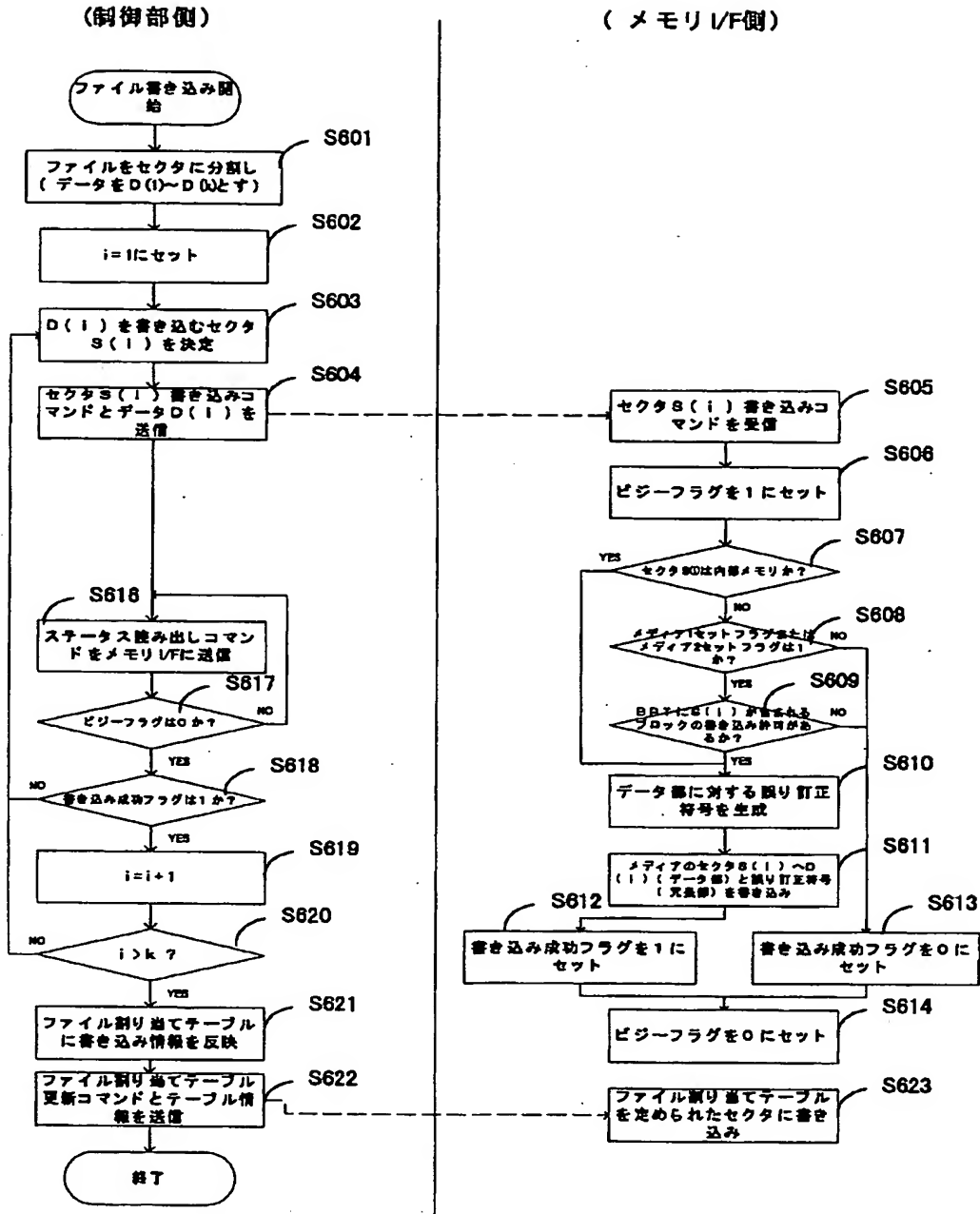
フロー 1-4 : 相互認証・鍵共有フロー (cont.)

【図 25】



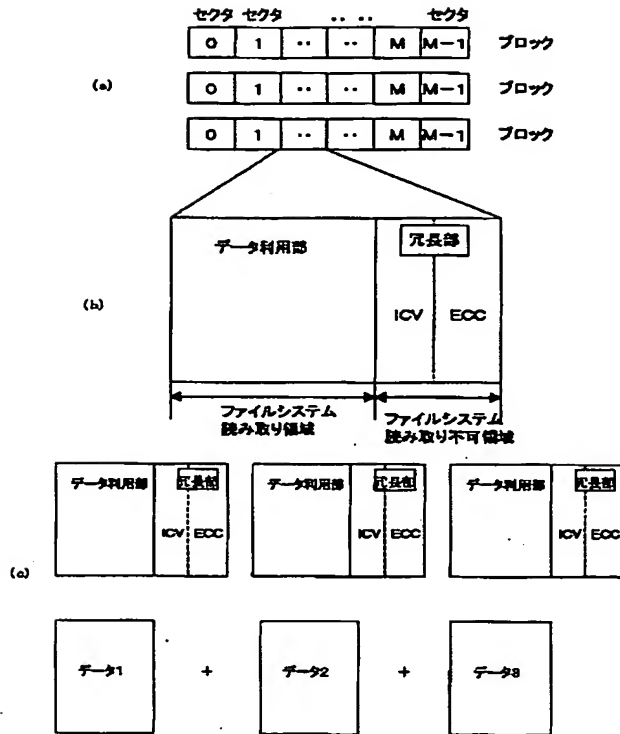
ファイルの読み出し処理

【図 26】

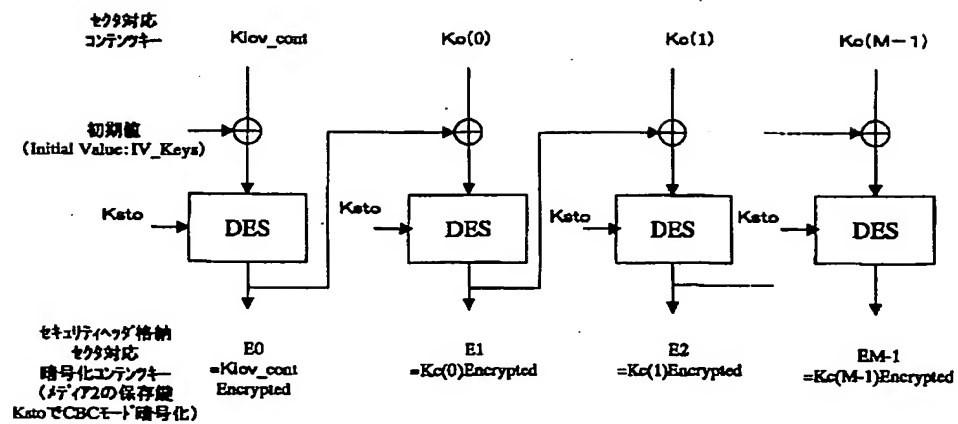


ファイルの書き込み処理

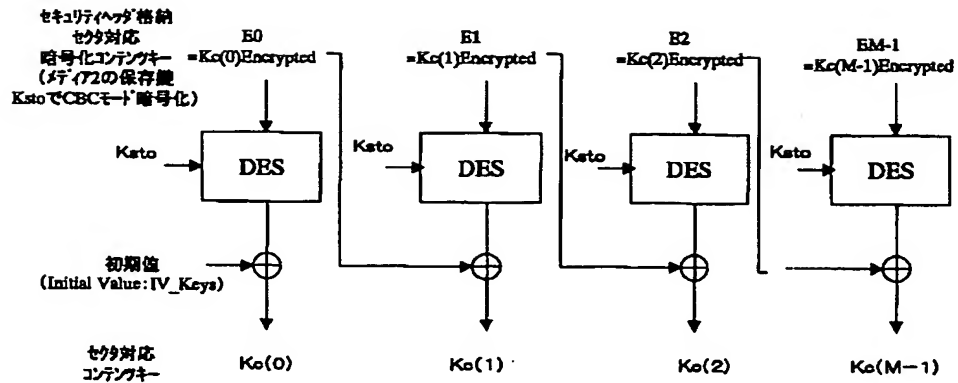
【図31】



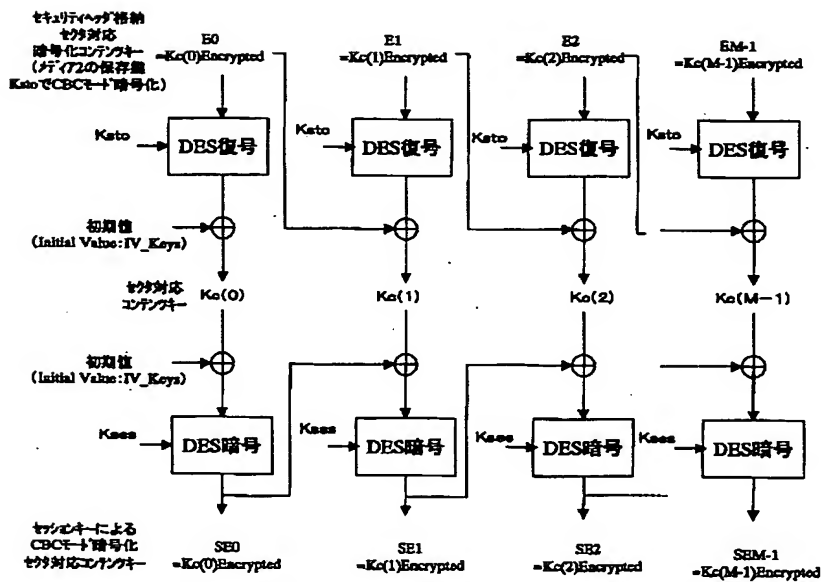
【図32】



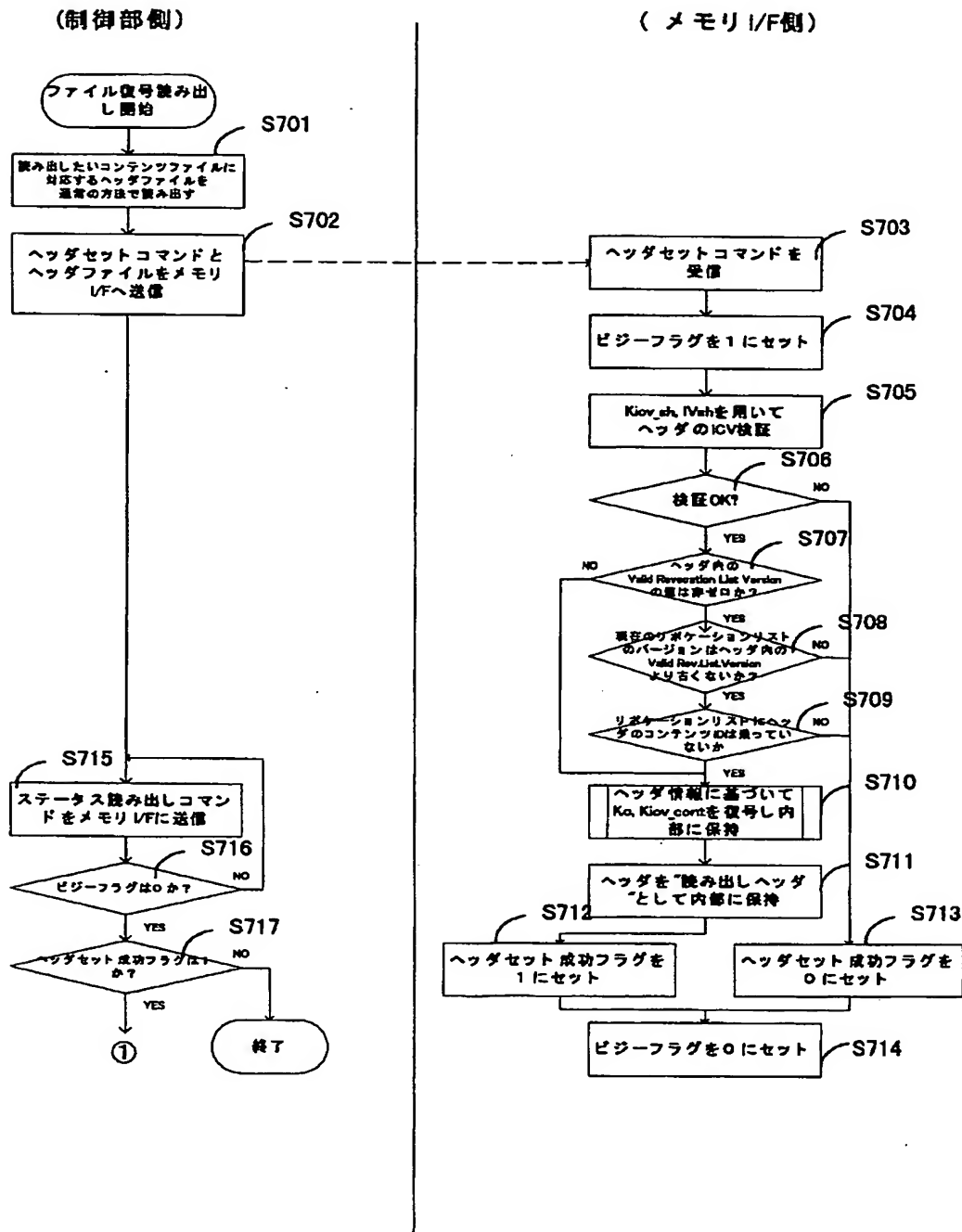
【図 33】



【図 34】

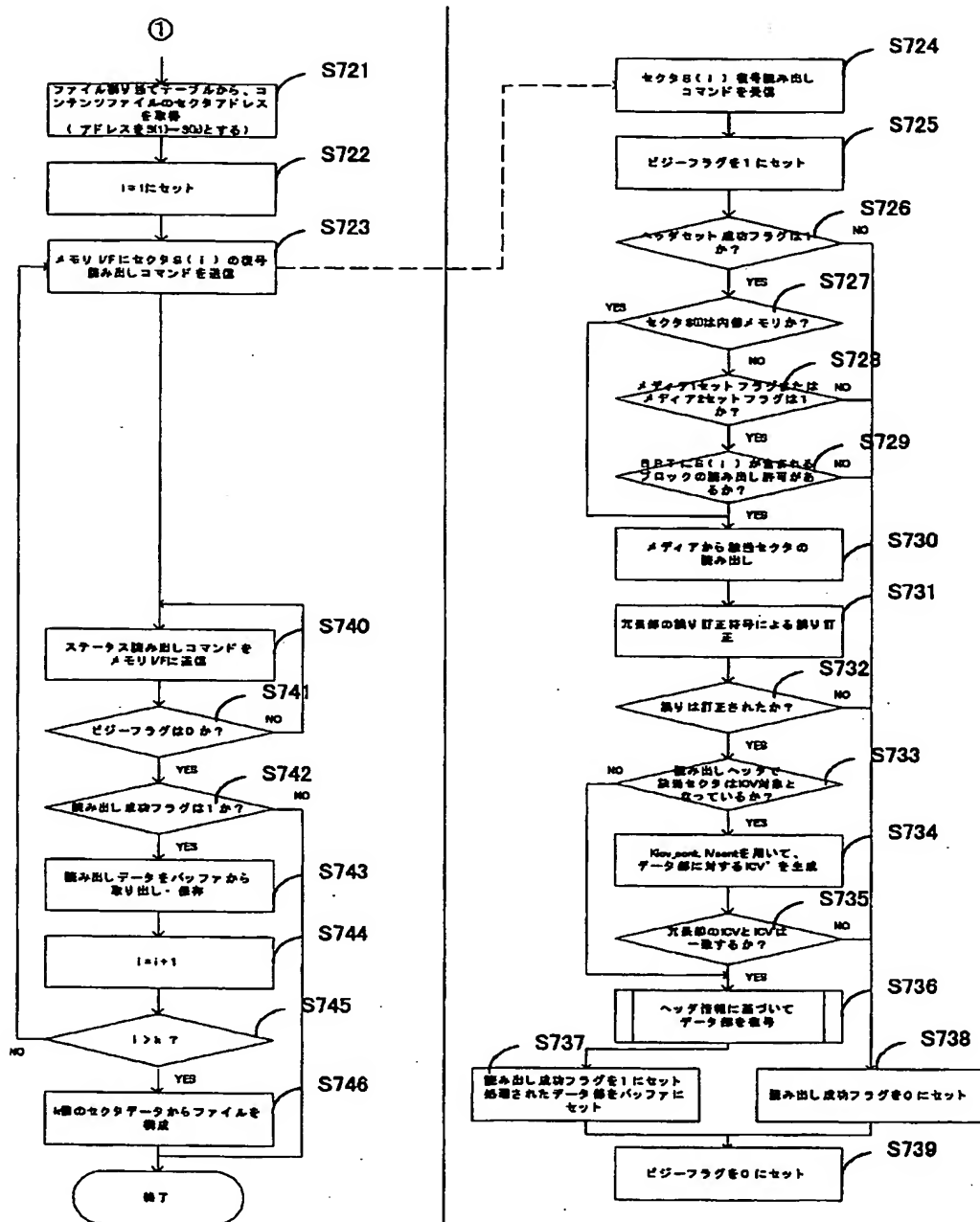


【図 35】



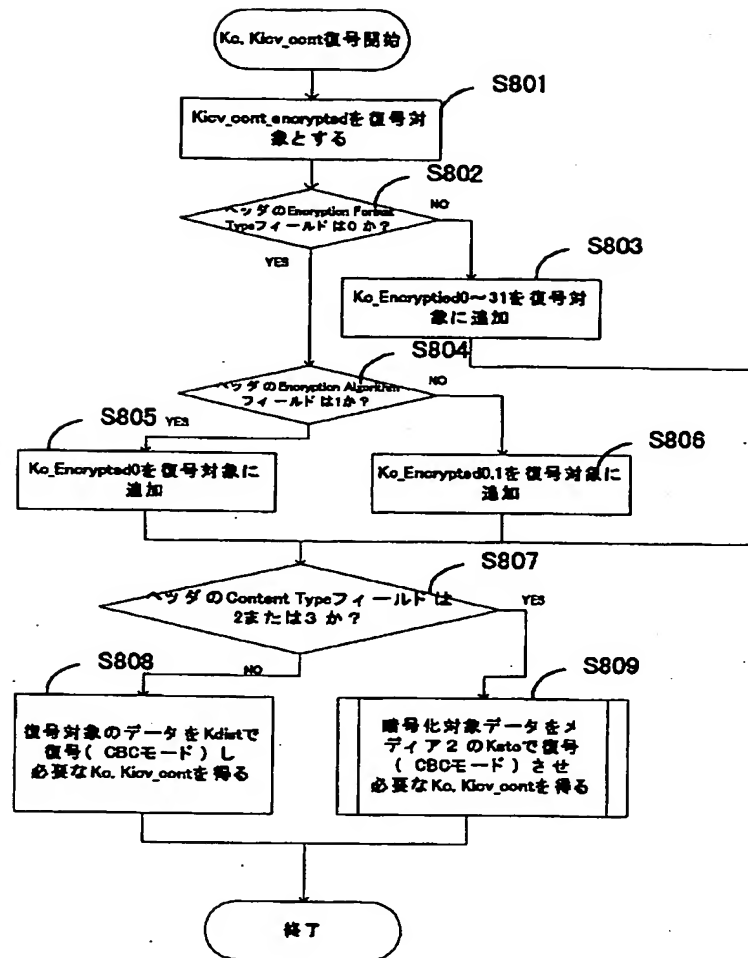
ファイルの復号読み出し処理

ファイルの復号読み出し処理



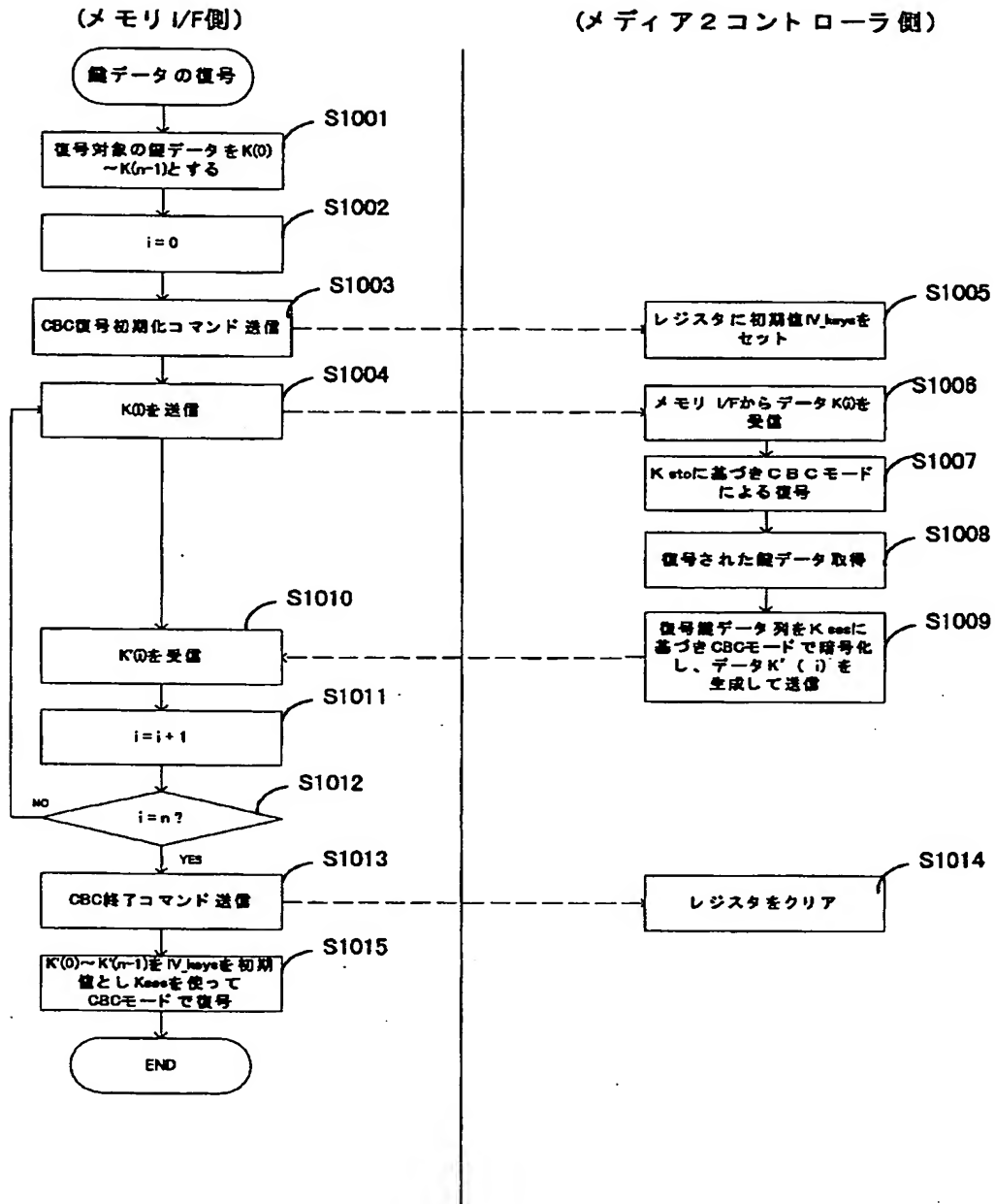
【図37】

(メモリ/F側)



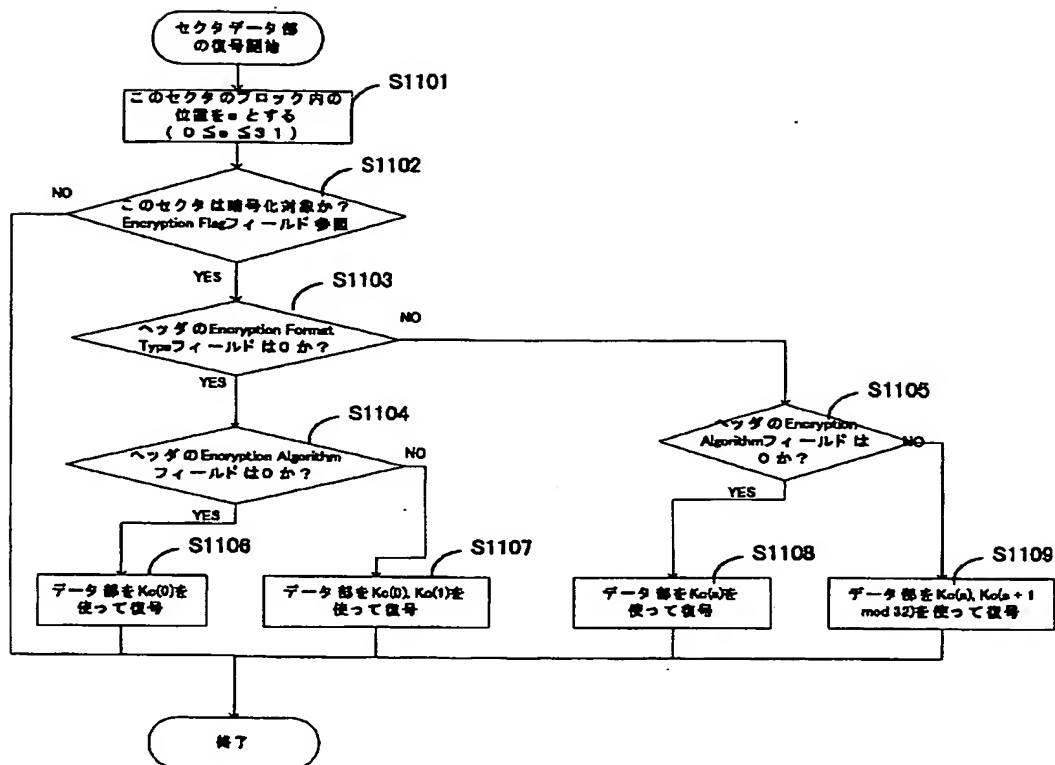
フロー 4-3 : Kc, Kiov_contの復号

【図38】



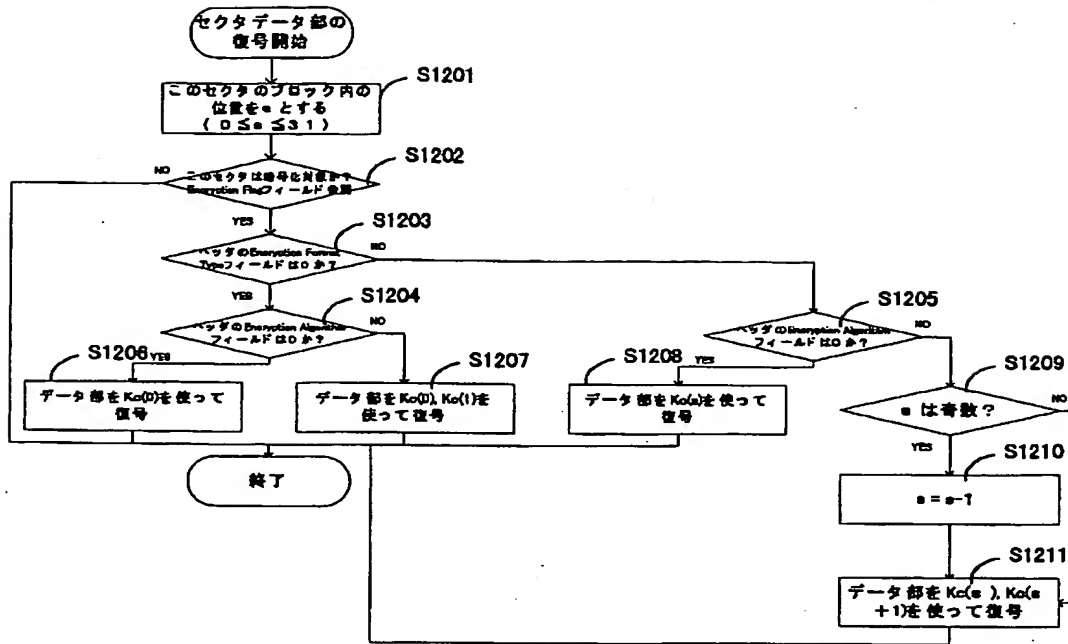
復号対象データをメディア 2 の Ksto で復号

【図39】



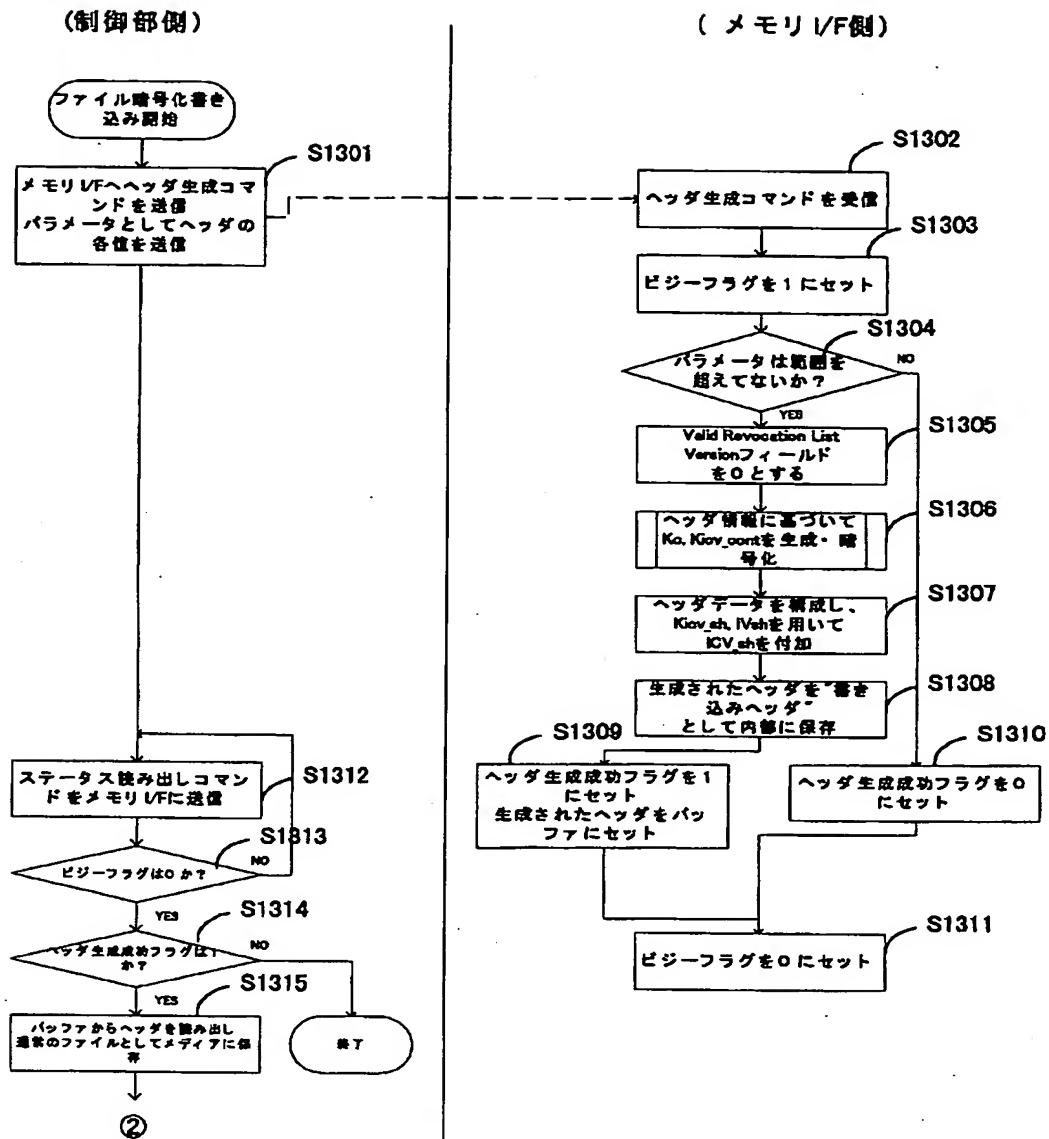
セクタデータ部の復号(その1)

【図40】



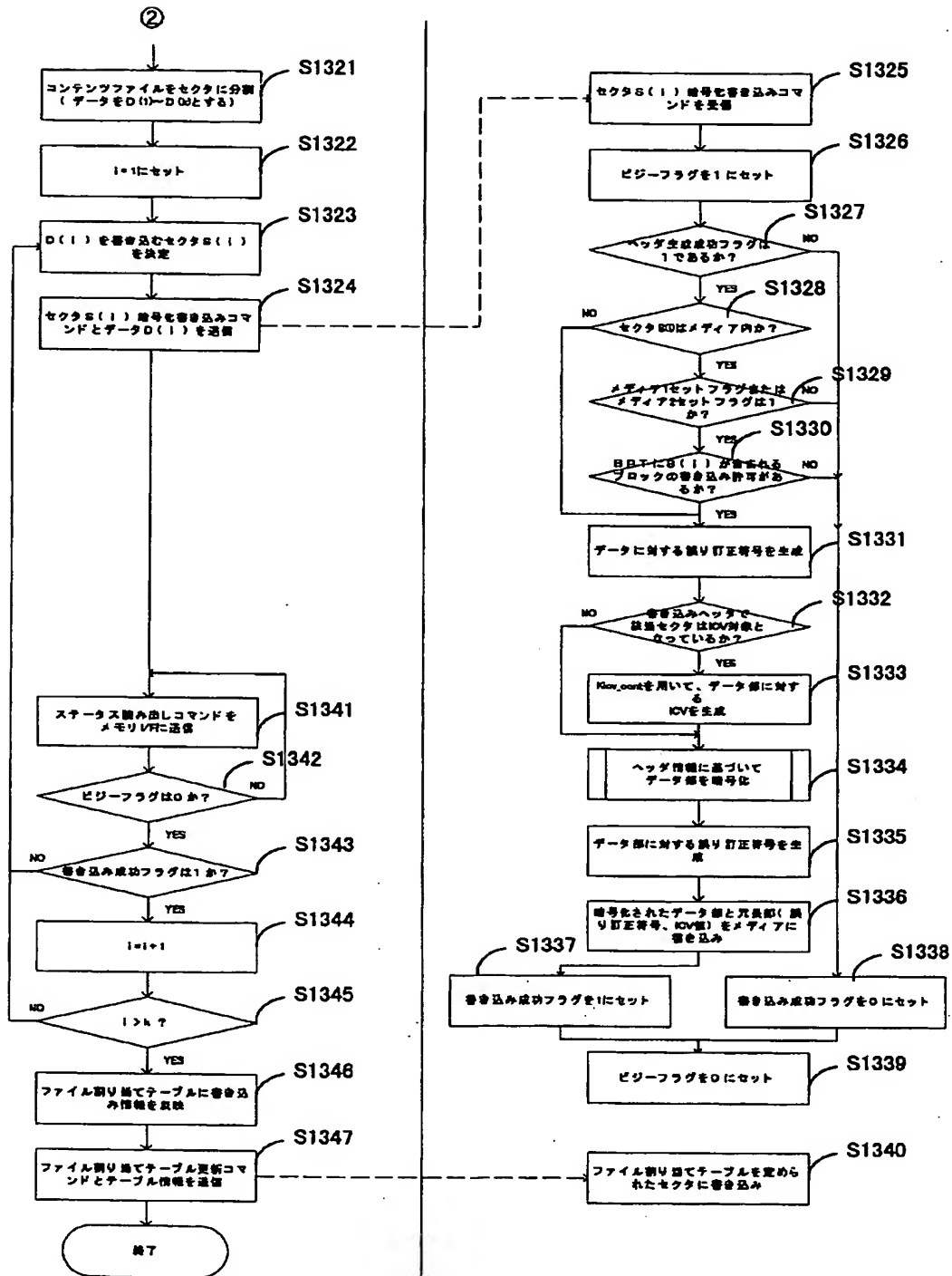
セクタデータ部の復号(その2)

【図 41】



ファイルの暗号化書き込み処理

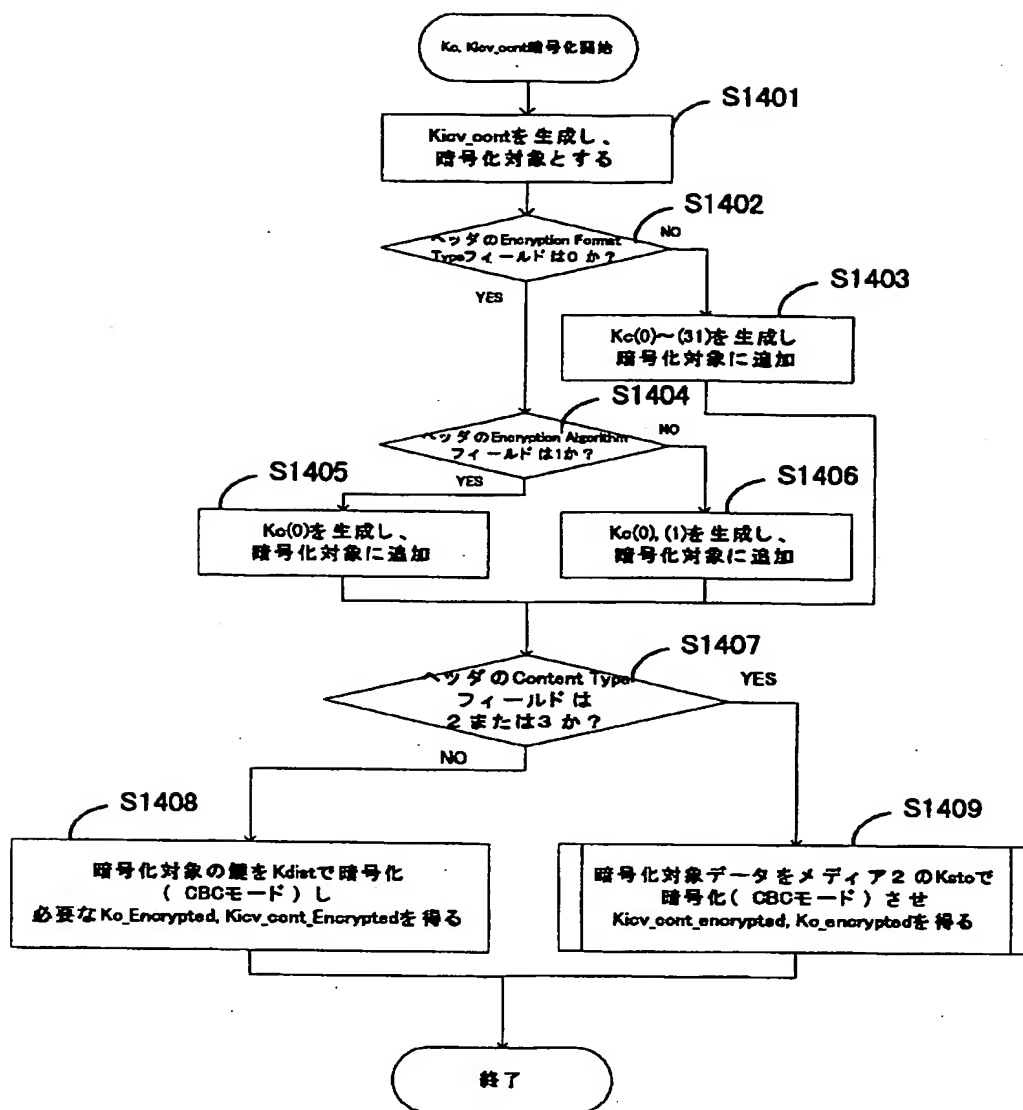
【図 42】



ファイルの暗号化書き込み処理

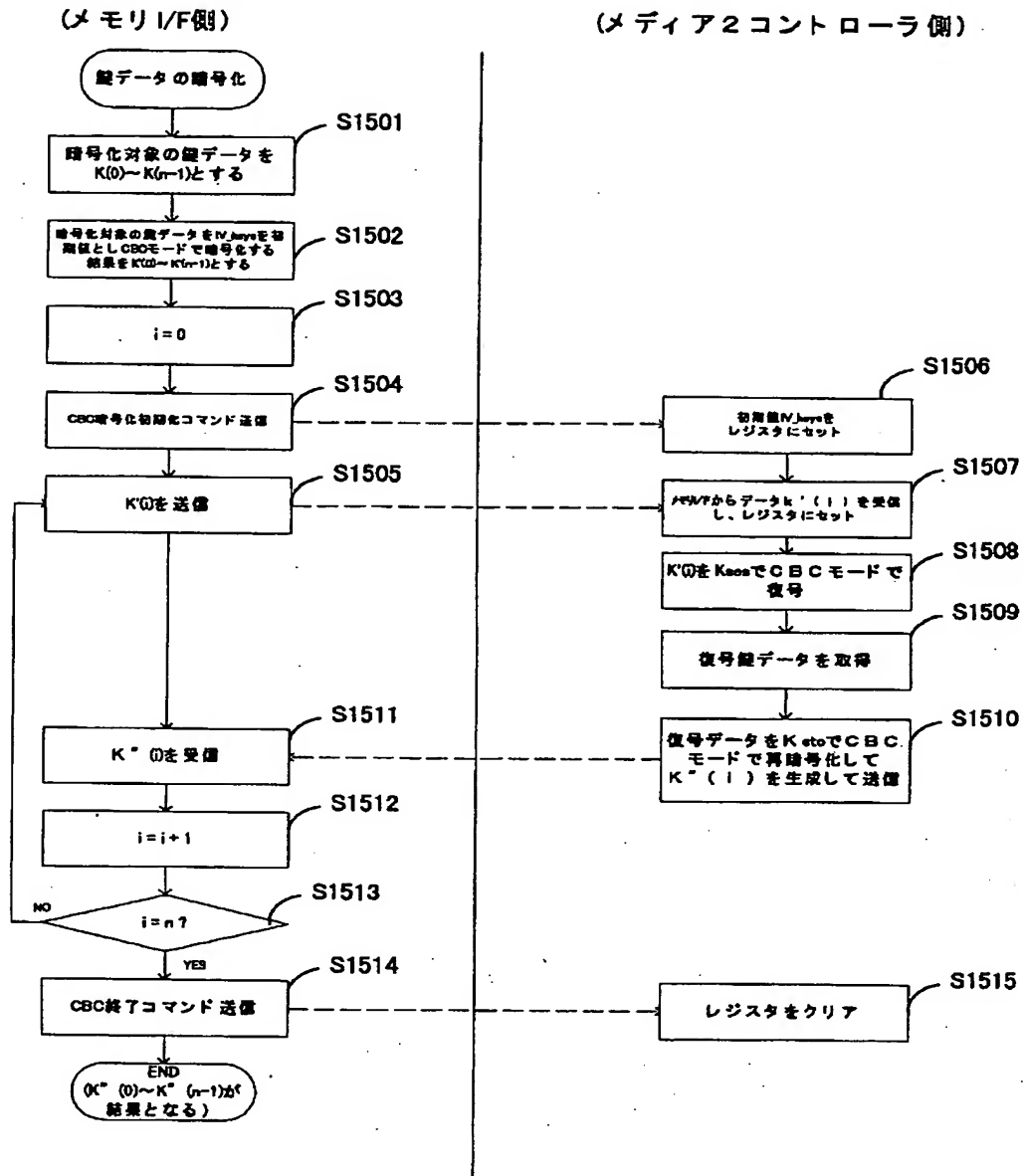
【図43】

(メモリ/F側)

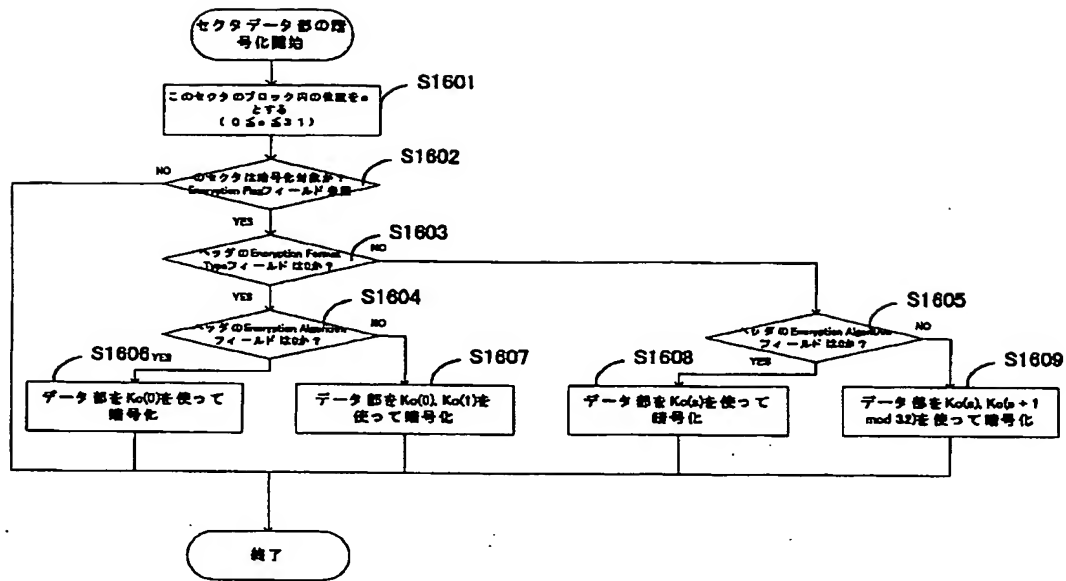


Kc, Kicv_contの暗号化

【図 44】

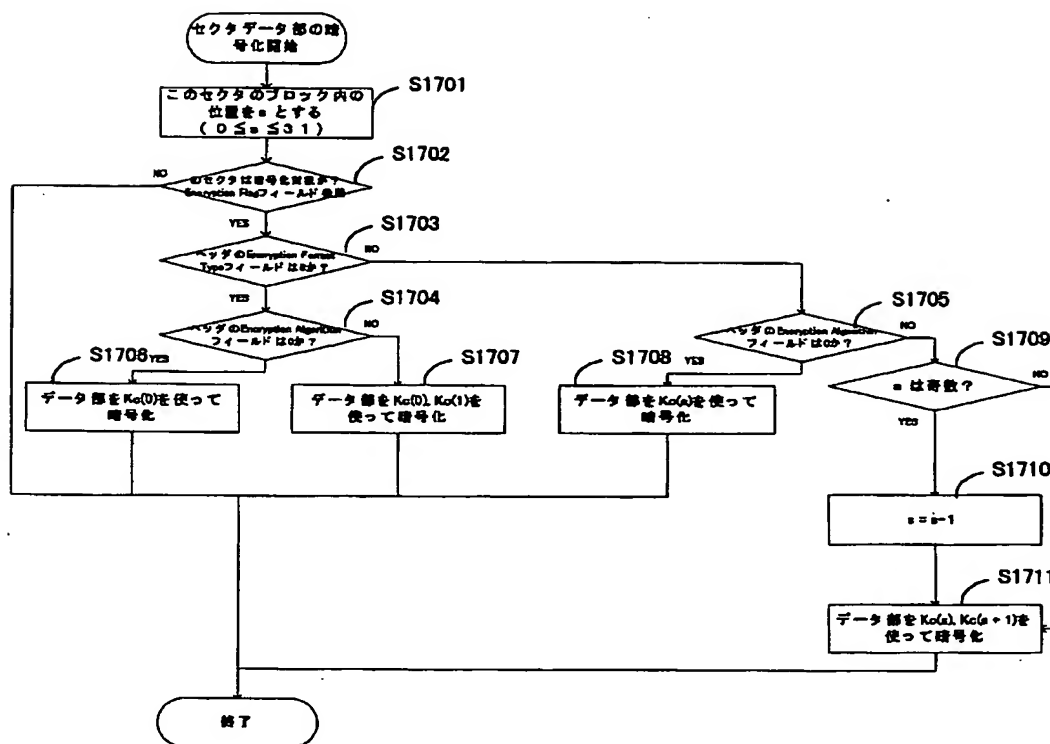
暗号化対象データをメディア 2 の $Ksto$ で暗号化

【図 45】



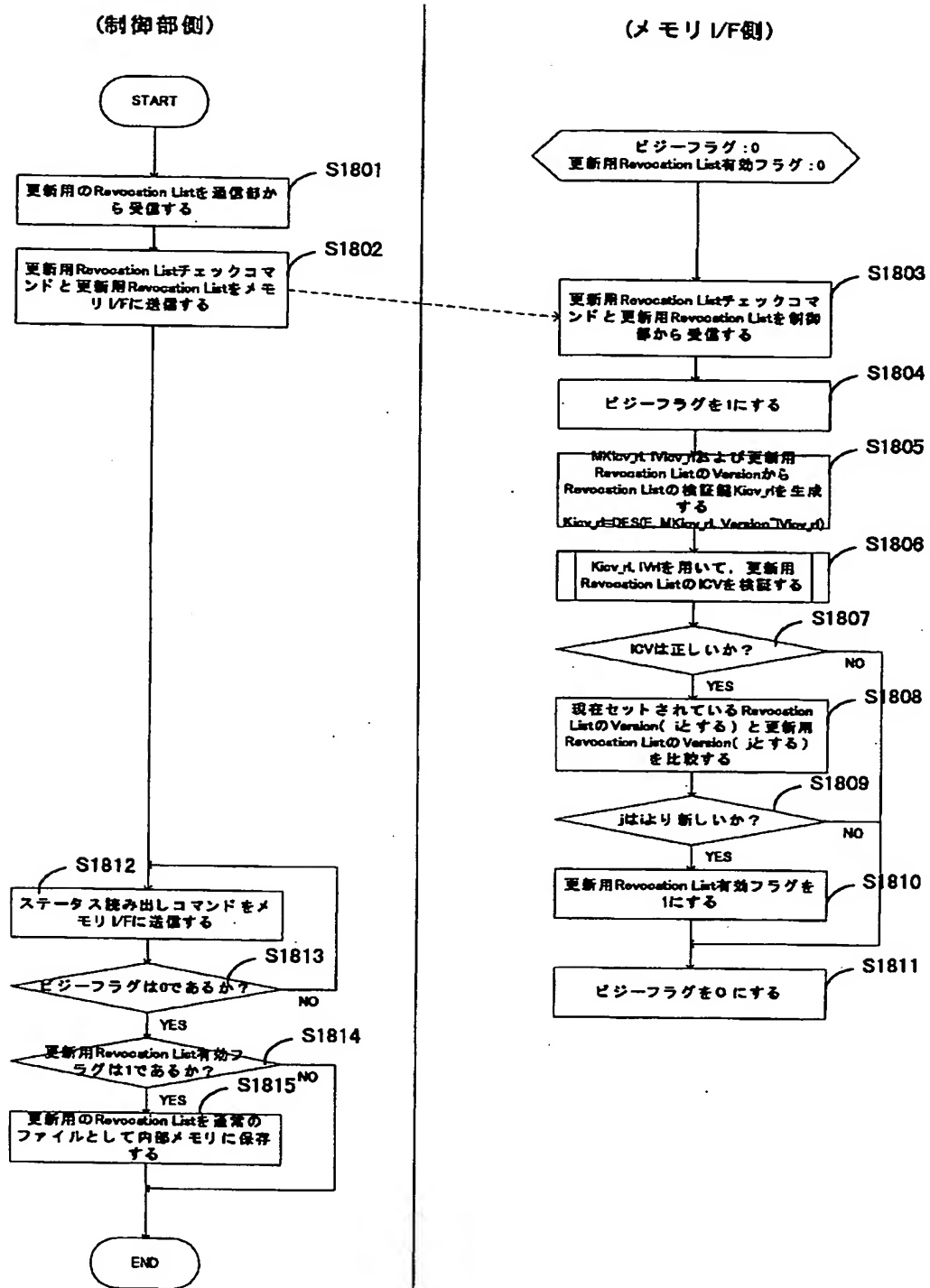
セクタデータ部の暗号化(その1)

【図 46】



セクタデータ部の暗号化(その2)

【図47】



Revocation Listの更新

フロントページの続き

(72)発明者 吉野 賢治
東京都品川区北品川 6 丁目 7 番35号 ソニ
ー株式会社内
(72)発明者 白井 太三
東京都品川区北品川 6 丁目 7 番35号 ソニ
ー株式会社内

Fターム(参考) 5B017 AA06 BB09 CA09 CA12 CA14
5C053 FA13 FA23 FA27 FA30 GB06
JA01 JA21 KA24 LA11 LA14
5D044 AB05 AB07 CC04 CC08 DE47
DE49 FG18 GK12 HH15 HL11